

Міністерство освіти і науки України

Національний університет водного господарства та
природокористування

Кафедра автоматизації, електротехнічних та комп'ютерно-
інтегрованих технологій

04-03-259М

Методичні вказівки

до виконання практичних робіт з навчальної дисципліни
«Теорія автоматизованого регулювання електроустановок»
для здобувачів вищої освіти першого (бакалаврського) рівня
за освітньо-професійною програмою «!Електроенергетика,
електротехніка та електромеханіка» спеціальності
141 «Електроенергетика, електротехніка та електромеханіка»
денної та заочної форм навчання

Рекомендовано науково-методичною
радою з якості ННІ АКОТ
Протокол № 01 від 11.11.2021 р.

Рівне – 2021

Методичні вказівки до виконання практичних робіт з навчальної дисципліни «Теорія автоматизованого регулювання електроустановок» для здобувачів вищої освіти першого (бакалаврського) рівня за освітньо-професійною програмою «Електроенергетика, електротехніка та електромеханіка» спеціальності 141 «Електроенергетика, електротехніка та електромеханіка» денної та заочної форм навчання [Електронне видання] / Стеценко А. М. – Рівне : НУВГП, 2021. – 81 с.

Укладач: Стеценко А. М., к.т.н., доцент кафедри автоматизації, електротехнічних та комп'ютерно-інтегрованих технологій.

Відповідальний за випуск: Древецький В. В., д.т.н., професор, завідувач кафедри автоматизації, електротехнічних та комп'ютерно-інтегрованих технологій.

Керівник групи забезпечення спеціальності 141 «Електроенергетика, електротехніка та електромеханіка»: Василець С. В., д.т.н., професор, професор кафедри автоматизації, електротехнічних та комп'ютерно-інтегрованих технологій.

© А. М. Стеценко, 2021

© НУВГП, 2021

Зміст

Робота №1. Розробка і випробування системи контролю та керування функціонуванням адміністративної будівлі (система типу «Розумний дім») на базі програмованого логічного контролера (ПЛК) Zelio Logic.

Робота №2. Розробка і випробування системи контролю та керування температурою повітря і вологості ґрунту в теплиці на базі ПЛК Zelio Logic та її випробування мовою LD.

Робота №3. Розробка і випробування системи контролю та керування температурою повітря і вологості ґрунту в теплиці на базі ПЛК Zelio Logic та її випробування мовою FBD.

Робота №4. Розробка і випробування системи контролю та керування на базі ПЛК Siemens Logo.

Робота №1. Розробка і випробування системи контролю та керування функціонуванням адміністративної будівлі (система типу «Розумний дім») на базі програмованого логічного контролера (ПЛК) Zelio Logic.

1. Мета роботи

Навчитися розробляти системи контролю та керування функціонуванням адміністративної будівлі (система типу «Розумний дім») на базі програмованого логічного контролера. Вивчити будову, правила підключення, характеристики, меню контролера Zelio Logic2. Навчитися програмувати контролер Zelio Logic2 мовою драбинкових діаграм (LD – Ladder Diagram).

2. Теоретичні відомості

2.1 Системи контролю та керування типу «Розумний дім»

Першим інтелектуальним будинком у світі став «Будинок трону» японського професора Кена Сакамури у Токіо, побудований наприкінці 1980-х років. Давачі погоди відкривали вікна, коли дув свіжий бриз, і вмикали кондиціонер, коли ставало жарко; якщо радіо грало надміру голосно, вікна автоматично закривалися, щоб не турбувати сусідів; якщо дзвонив телефон, комп'ютер знижував звук аудіо системи, і так далі.

Основою «розумного» керування інженерними системами і системами життєзабезпечення являється технологія, що дозволяє здійснювати інтелектуальне керування, регулювання та гнучку оптимізацію систем:

- керування опаленням: керування системами опалення різних типів, економія витрат на опалення;
- водопостачання: контроль протікання води у всіх приміщеннях, керування системами водо підготовки;
- газопостачання: контроль витікання газу;
- охорона: контроль проникнення у приміщення, периметральний контроль, імітація присутності людей, імітація присутності тварин, обмеження доступу, відеонагляд;
- пожежна безпека: контроль займання;

- керування електроенергією: контроль електромереж, економія затрат;
- керування мікрокліматом у приміщенні: керування температурою, відносною вологістю повітря, контроль вмісту небезпечних газів;
- оповіщення: голосове оповіщення, дзвінок на мобільні та міські номери, sms, Internet;
- освітлення: зв'язок із присутністю людей, керування за часом, сценарне керування, сценарії зовнішнього освітлення;
- будильник: голос, телефон, мелодії;
- голосовий пейджер: запис та відтворення повідомлень;
- басейн: керування нагрівом, керування фільтрацією;
- полив: автоматичний полив;
- жалюзі: автоматичне відкриття-закриття жалюзі та воріт, сценарне керування жалюзі;
- керування побутовими пристроями: керування усіма пристроями з одного пульта дистанційного керування, зв'язок між приміщеннями, вхідний дзвінок з широким вибором мелодій.

2.2 Технічні засоби для реалізації системи «Розумний будинок»

Для реалізації системи «Розумний дім» використовуються різноманітні технічні засоби: датчики, контролери, пульти та панелі керування, домофони, GSM модеми, виконавчі механізми і регулюючі органи. Для реалізації функцій керування використовуються прості за набором функцій програмовані контролери із контролерного ряду тієї чи іншої фірми-виробника: Zelio Logic (Schneider Electric), Logo (Siemens), Mitsubishi (Mitsubishi Electric), Forth Logic (Електросвіт) та інші. Розглянемо детальніше програмований контролер Zelio Logic, на базі якого і будемо реалізовувати систему керування адміністративною будівлею.

2.2.1 Будова контролера Zelio Logic2

Загальний вигляд контролера Zelio Logic2 представлено на рис. 1.1.



Рис. 1.1 Загальний вигляд контролера Zelio Logic2

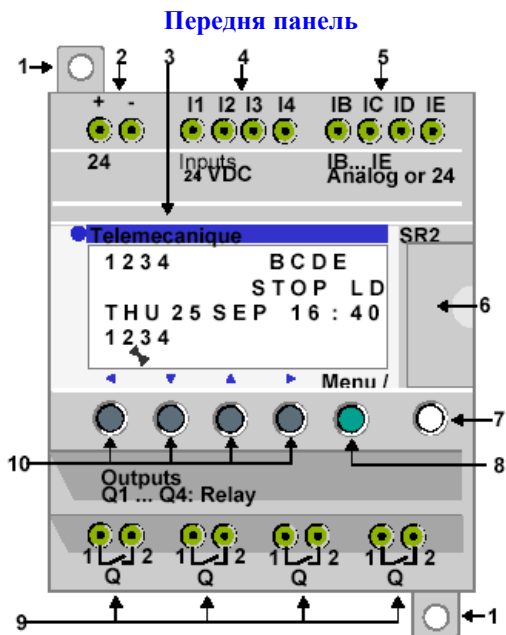


Рис. 1.2 Вигляд передньої панелі контролера Zelio Logic2

Таблиця 1.1

Позиція	Опис
1	Висувна монтажна ніжка
2	Гвинтові клеми для підключення живлення
3	Рідкокристалічний дисплей (4 рядки, 18 символів у кожному)
4	Гвинтові клеми для дискретних входів
5	Гвинтові клеми для аналогових входів (0...10 В), які можуть використовуватися як дискретні у деяких моделях
6	Роз'єм для резервної пам'яті або кабеля під'єднання ПК
7	Клавіша Shift
8	Клавіша вибору і підтвердження
9	Гвинтові клеми дискретних виходів
10	Клавіши-стрілки (або сконфігуровані Z-клавіши)

Дисплей

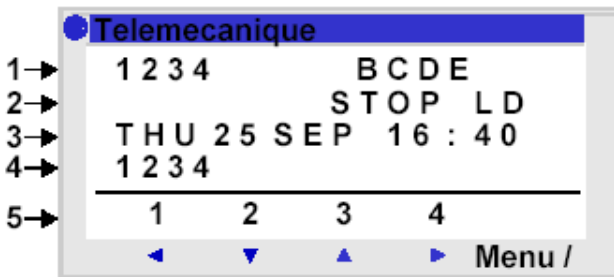


Рис. 1.3 Дисплей контролера Zelio Logic2

Таблиця 1.2

Позиція	Опис
1	Відображення стану входів (B...E відображають аналогові входи)
2	Відображення режиму роботи (RUN / STOP) та режиму програмування (LD / FBD)
3	Відображення дати (число і час для пристроїв, що підтримують таку можливість)
4	Відображення стану виходів

2.2.2 Характеристики та підключення

Підключення контролера постійного струму

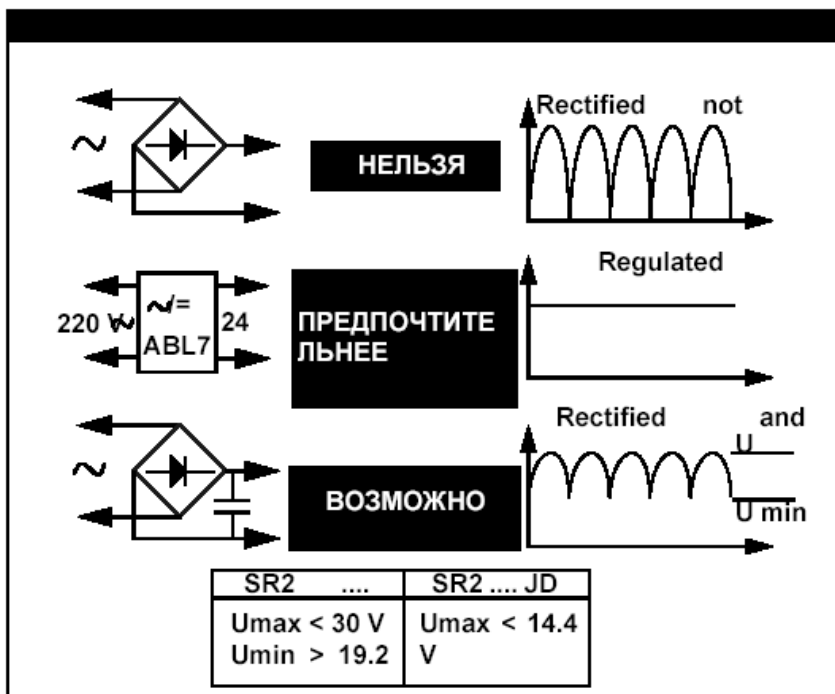


Рис. 1.4 Правила підключення контролера постійного струму

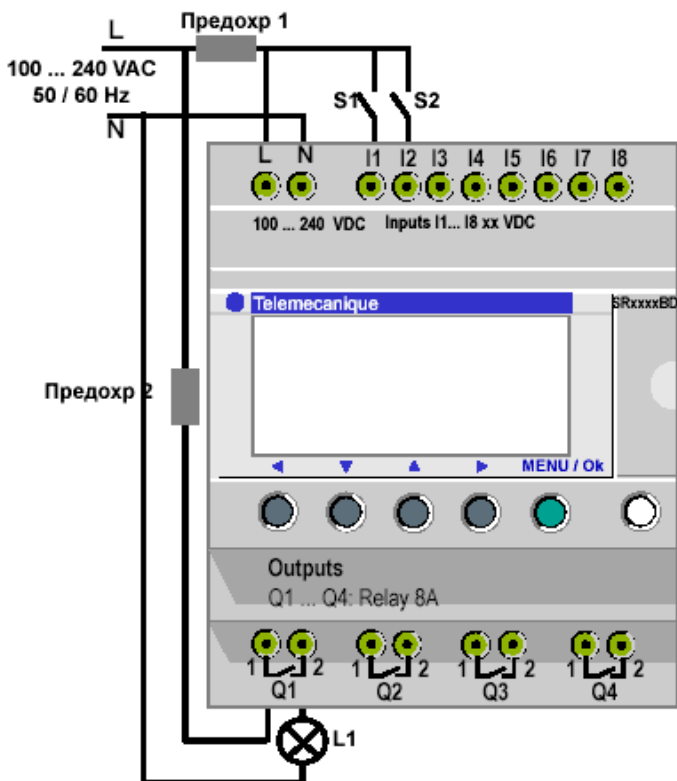


Рис. 1.5 Приклад підключення 2-х вимикачів до входів I1 та I2 та лампочки до виходу Q1 контролера

2.2.3 Меню

Головне меню

Таблиця 1.3

№	Назва	Призначення
1	Programming	Введення драбинкових діаграм.
2	Parameters	Введення та зміна параметрів елементів програми.
3	Monitoring	Доступний тільки в режимі LD / RUN. Даний режим використовується для динамічного відображення стану входів-виходів контролера. Користувач може динамічно змінювати значення параметрів функцій, якщо

		вони розблоковані (клавіши Shift + Param).
4	RUN / STOP	Запуск чи зупинка програми. Коли здійснюється перемикання від STOP до RUN, програма ініціалізується.
5	Configuration	Меню конфігурації.
6	Clear Program	Видалення програми.
7	Transfer	Передача програми у модуль резервної пам'яті або навпаки.
8	Language Menu	Меню вибору мови.
9	Version	Визначення версії системних компонентів: вбудованого програмного забезпечення, FBD-функції, LD-функцій.
10	Fault	Меню помилок, яке дозволяє відображати номери помилок або попереджень, виявлених програмним забезпеченням контролера.

Меню конфігурації – Configuration

Таблиця 1.4

№	Назва	Призначення
1	Password	Меню введення пароля.
2	Filter	Вибір швидкості визначення зміни стану всіх дискретних входів.
3	Zx-Keys	Дозволяє користувачу дозволити або заборонити використання клавіш курсору у якості кнопок. Доступно тільки в режимі LD.
4	Change D/T	Задання дати та часу для модулів з годинником.
5	Change Summ/Wint	Автоматична зміна часового діапазону літо-зима для модуля з годинником.
6	Watchdog Cycle	Задання тривалості циклу виконання програми.

2.2.4 Програмування контролера Zelio Logic2 мовами драбинкових діаграм (LD) та FBD-блоків

Контролер Zelio Logic2 можна програмувати за допомогою командних клавіш (мова LD) та за допомогою спеціалізованого програмного забезпечення Zelio Soft2 (мови LD та FBD).

Розглянемо правила програмування контролера Zelio Logic2 за допомогою спеціалізованого програмного забезпечення Zelio Soft2.

2.2.4.1 Створення нової програми

Для того, щоб створити нову програму необхідно виконати наступні дії.

1. Вибрати меню **File/New** або натиснути піктограму **Create a new program** під час запуску програми Zelio Soft2. В результаті з'явиться вікно вибору типу модуля.

2. Виберіть потрібний модуль за допомогою ЛК миші. Модулі згруповані за такими ознаками:

- кількість входів-виходів;
- наявність або відсутність дисплею;
- можливість під'єднання шасі розширення.

В результаті здійснення вибору типу модуля з'являється весь список модулів даного типу у вигляді таблиці.

3. Виберіть модуль подвійним натисканням миші або натисніть клавішу **Далее (Next)**. На цьому етапі може виникнути такі ситуації:

- модуль не підтримує можливості розширення та програмується лише мовою LD – перейдіть до кроку 7;
- модуль не підтримує можливості розширення і програмується мовами LD та FBD – перейдіть до кроку 6;
- модуль підтримує розширення, тоді на екрані з'явиться інформація про тип модуля, вибраний у попередньому пункті та типи можливих розширень. Натиснувши кнопку **Add**, можна вибрати необхідне розширення. Для переходу до наступного кроку натисніть кнопку **Далее**. Модуль розширення при необхідності можна видалити кнопкою **Delete**.

4. Виберіть необхідний модуль розширення.

5. Підтвердіть вибір (кнопка **Далее**). В результаті з'явиться вікно вибору мови програмування.

6. Виберіть потрібну мову (LD або FBD). Натисніть кнопку **Далее (Next)**.

7. З'явиться вікно редагування програми.

2.2.4.2 Мова драбинкових діаграм (Ladder Diagram Language)

Загальні положення

Програма, написана мовою LD, складається з мереж з'єднаних між собою елементів. Виконання програми відбувається згідно географічного розміщення елементів зверху донизу і зліва направо.

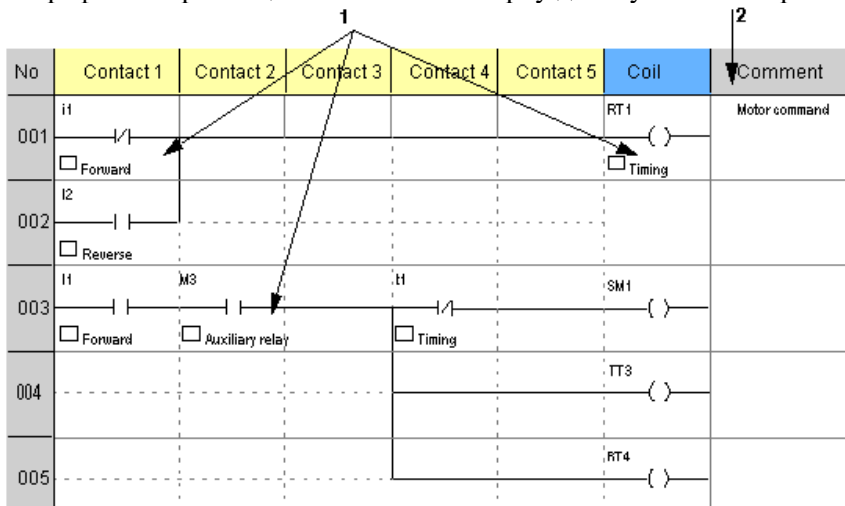


Рис. 1.6 Приклад програми, написаної мовою LD

Таблиця 1.5

№	Елемент	Функція
1	Графічні елементи	<ul style="list-style-type: none"> - Входи-виходи контролера (кнопки, давачі, реле, світлові індикатори тощо). - Контакти функціональних блоків (таймерів, лічильників тощо). - Логічні операції. - Внутрішні змінні (допоміжні реле) контролера.
2	Коментар	Для кожної лінії драбинкової діаграми або конкретного графічного елемента (є необов'язковим).

Вікно створення програми поділяється на декілька частин (рис. 1.7).

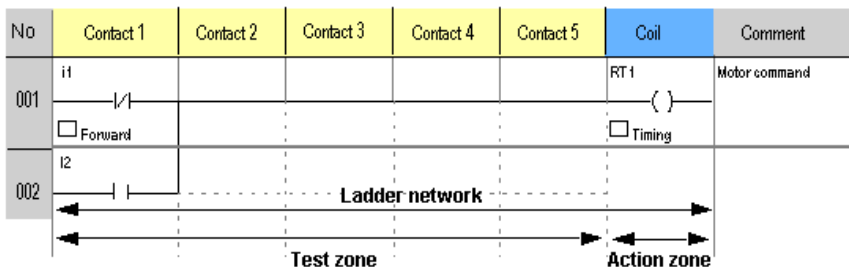


Рис. 1.7 Структура вікна створення програми мовою LD

Кожен рядок LD-програми складається з набору графічних елементів, які розміщені у прямокутниках сітки:

- максимальна кількість рядків – 120;
- кожен рядок може містити до 5 контактів і 1 котушку.

LD-діаграма поділена на 2 зони:

- **test zone**, в якій розміщуються контакти;
- **action zone**, в якій розміщуються котушки (відображає результат операцій).

Графічні елементи у мові LD

1. Контакти

Дані елементи розміщуються у test zone і займають 1 чарунку.

Таблиця 1.6

Назва	Графічне представлення (LD)	Графічне представлення (Electrical)	Функції
Нормальний відкритий контакт			Замикається, коли асоційований з ним елемент (вимикач, давач), активний.
Нормальний замкнутий контакт			Розмикається, коли асоційований з ним елемент (вимикач, давач), активний.

2. Елементи зв'язку

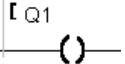
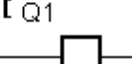
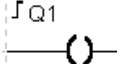
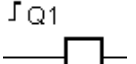
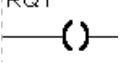

Використовуються для зв'язку між елементами, що розміщені у test-зоні або action-зоні.

Таблиця 1.7

Назва	Графічне представлення	Функції
Горизонтальний зв'язок	—	Використовується для зв'язку між послідовно розміщеними елементами.
Вертикальний зв'язок		Використовується для зв'язку між паралельно розміщеними елементами.

3. Котушки

Таблиця 1.8

Назва	Графічне представлення (LD)	Графічне представлення (Electrical)	Функції
Пряма котушка			На котушку подається живлення, якщо контакти, до яких вона під'єднана, активні.
Імпульсна котушка			На котушку подається живлення, якщо контакти, до яких вона під'єднана, змінюють свій стан.
Замкнена котушка (Set)			На котушку подається живлення, якщо контакти, до яких вона під'єднана, активні. Її стан не зміниться навіть якщо її контакти розімкнуться.
Розімкнена котушка (Reset)			Котушку стає неактивною, якщо контакти, до яких вона під'єднана, активні. Вона зберігає свій стан незалежно від подальшої поведінки її контактів.

Увага! Котушки Set і Reset повинні використовуватися разом. Функція Reset має пріоритет перед функцією Set. Кожен з виходів Q повинен використовуватися у програмі тільки 1 раз, крім котушок Set і Reset.

Етапи створення LD-програми

1. Складіть список входів-виходів і присвойте їм коментарі.
2. Складіть список необхідних керуючих функцій (підрахунок кількості машин, дискретне керування).
3. Реалізуйте кожну функцію за такою схемою.

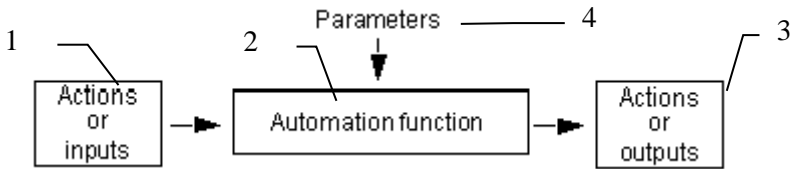


Рис. 1.8 Схема реалізації керуючої функції

- 1 – керуючі дії або входи;
- 2 – керуюча функція;
- 3 – керуючі дії або виходи;
- 4 – параметри.
4. Присвойте коментар кожній функції.
5. Протестуйте кожну функцію, використовуючи режим емуляції.

Режими програмування

1. Режим Zelio (Zelio Entry Mode)

Даний режим дозволяє створити програму мовою LD за допомогою емуляції використання командних кнопок на передній панелі контролера.

Приклад:

```

i1●-----C1-----TX1
z1-----┘
I2-----iB-----[Q1
H1----------[M1
  
```

Рис. 1.9 Приклад LD-програми у режимі Zelio

Даний режим зручний тим, що надає програмісту можливість навчитися програмувати контролер через командні кнопки у середовищі Zelio Soft, не маючи самого контролера.

2. Вільний режим (Free Entry Mode)

Даний режим дозволяє засвоїти прийоми створення програм за допомогою самого середовища програмування без імітації передньої панелі контролера, а саме:

- використання панелей інструментів;
- створення програм за допомогою перетягування елементів у потрібне місце драбинкової діаграми;
- використання вікон параметрів елементів;
- можливість переглядати усю програму у цілому.

Приклад:

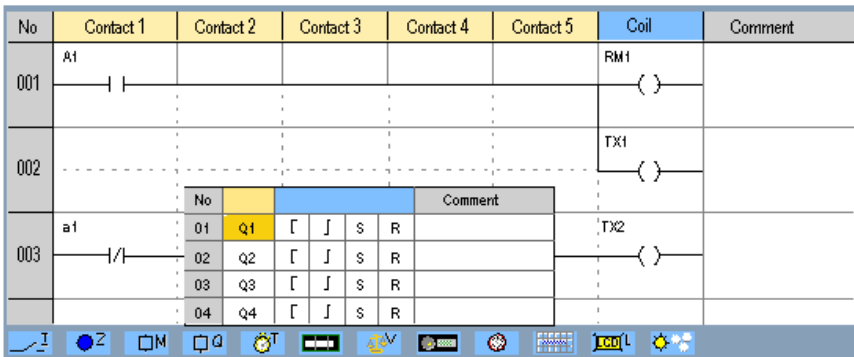


Рис. 1.10 Приклад LD-програми у вільному режимі

Даний режим надає іще ряд додаткових переваг:

- вибір типу представлення символів (LD, електричні аналоги);
- можливість додати коментар до рядка.

3. Параметричний режим (Parameter Mode)

Даний режим дозволяє представити усі функції керування, які використовуються у програмі, з їх параметрами у вигляді списку, який має такі позиції:

- номер за порядком;
- функція: таймер, лічильник тощо;
- мітка, що визначає функціональний блок;

- тип;
- уставка;
- вказівник замикавання;
- коментар.

Приклад:

Zelio entry Free entry Parametering Text entry						
No	Function	Label	Type	Preset	Lock	Comment
001	Counter	C1		C1 = 00001	No	Number of vehicles
002	Clock	⌚ 1			No	Opening time
003	Analog	A1	5: 7.0 <= 18	R = 7.0V	No	Primary circuit, voltage
004	Text Block	X1			Yes	Current counter value

Рис. 1.11 Список функціональних блоків, використаних у LD-програмі, при використанні параметричного режиму

Параметри конкретного функціонального блоку можна змінити, якщо двічі натиснути мишкою на рядку, де він описаний.

4. Текстовий режим (Text Entry Mode)

Даний режим дозволяє представити усі входи-виходи, які використані у програмі, у вигляді списку. Важливо присвоїти кожному входу та виходу зрозумілий коментар для того, щоб зробити програму зрозумілішою.

Приклад:

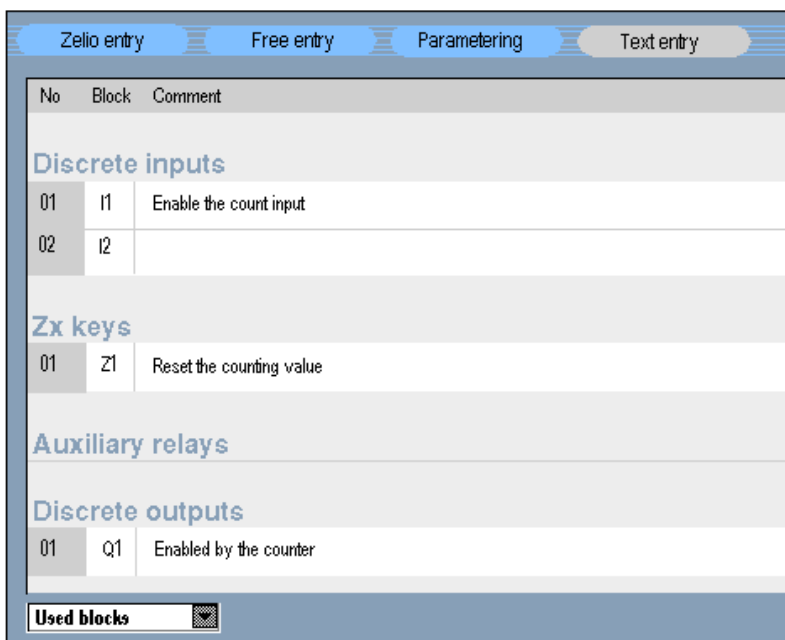


Рис. 1.12 Представлення входів та виходів у текстовому режимі

Введені коментарі будуть відображатися під графічними зображеннями елементів у драбинковій діаграмі (рис. 1.7).

Елементи мови драбинкових діаграм (LD)


Таблиця 1.9

№	Піктограма у Zelio Soft	Назва
1		Дискретні входи (Discrete Inputs)
2		Дискретні виходи (Discrete Outputs)
3		Входи-виходи Modbus (Modbus Inputs-Outputs)
4		Допоміжні реле Auxiliary Relays)
5		Z-клавіши (Zx Keys)
6		Лічильники (Counters)

7		Функціональний блок порівняння лічильника (Counter Comparator)
8		Швидкий лічильник (Fast Counter)
9		Годинники (Clocks)
10		Функціональний блок переходу на зимовий та літній час (Change to Summer/Winter Time)
11		Таймери (Timers)
12		Аналогові компаратори (Analog Comparators)
13		Функціональний блок Текст (Texts)
14		Функціональний блок підсвітки екрану (LCD Screen Backlighting)

Автоматична перевірка LD-програми

Середовище програмування Zelio Soft2 дозволяє здійснювати емуляцію програм, їх покрокову перевірку та налагодження. Якщо

знайдена якась помилка, піктограма , що розташована у верхній частині вікна редагування програми, змінює свій колір з блакитного на червоний.

Приклад:

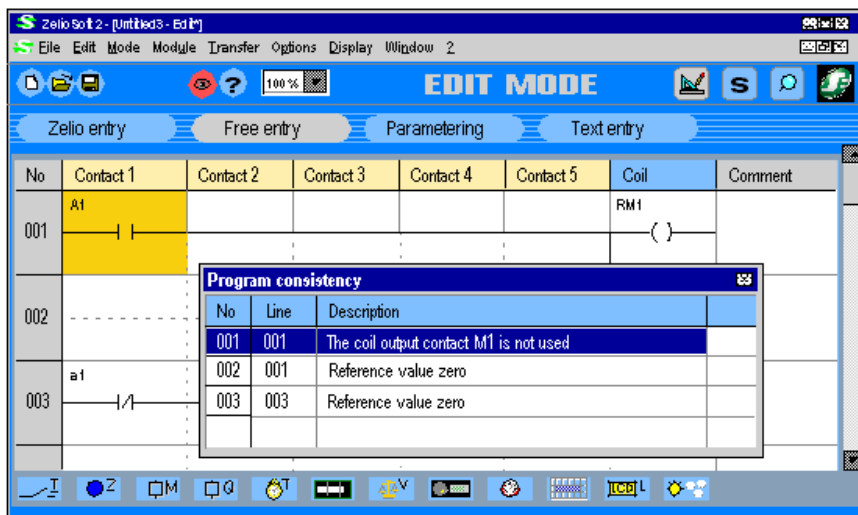


Рис. 1.13 LD-програма у режимі налагодження

Натиснувши на піктограму “Око”, можна відкрити діалогове вікно, яке містить наступну інформацію:

- номер помилки;
- місцезнаходження помилки: рядок або стовпчик;
- опис помилки.

Подвійне натискання мишки на назві помилки призводить до підсвітки конкретного елемента програми, де міститься помилка.

Увага! Навіть якщо символ “Око” став червоним, програма може бути запущена на емуляцію. Елементи, розміщені у програмі з допущенням помилок (відсутнє з’єднання), будуть неактивні.

Емуляція та моніторинг LD-програми

Функція емуляції використовується для виконання програми безпосередньо в Zelio Soft2 (у реальному часі).

Приклад:

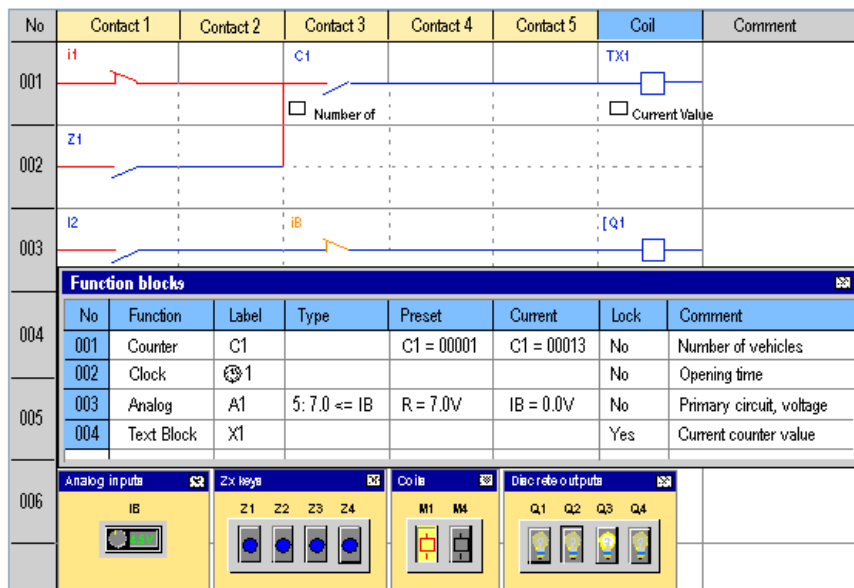


Рис. 1.14 LD-програма у режимі емуляції

Таблиця 1.10

Піктограма	Функція
	Перехід у режим емуляції
	Перехід у режим редагування
	Перехід у режим моніторингу
	Запуск програми на виконання
	Зупинка виконання програми
	Пуза/Запуск: зупиняє або перезапускає програму (активна тільки у режимі Run)
	Емуляція збою енергопостачання (активна тільки у режимі Run)

Функція моніторингу використовується для виконання програми в контролері (у режимі реального часу) і одночасної візуалізації в

Zelio Soft 2. Потрібно, щоб на комп'ютері і в контролері була запущена та сама програма. У даному режимі користувач може:

- відображати значення різних параметрів у вікні функціональних блоків;
- натиснути на блок, щоб змінити його параметри.

3. Програма роботи

1. Вивчити будову та характеристики контролера Zelio Logic 2.
2. Дослідити правила підключення контролера Zelio Logic 2.
3. Навчитися користуватися меню контролера Zelio Logic 2
4. Навчитися програмувати контролер Zelio Logic 2 мовою драбинкових діаграм (LD) за допомогою командних клавіш передньої панелі.
5. Навчитися програмувати контролер Zelio Logic 2 мовою драбинкових діаграм (LD) у середовищі Zelio Soft 2.
6. Навчитися записувати програми у контролер.
7. Виконати завдання, поставлені викладачем.

4. Опис лабораторного обладнання

1. Персональний комп'ютер.
2. Операційна система Windows.
3. Лабораторний стенд з логічним контролером Zelio Logic.
4. Середовища програмування Zelio Soft, Zelio Soft 2.

5. Порядок виконання роботи

1. Напишіть програму для програмованого логічного контролера (ПЛК) Zelio Logic мовою драбинкових діаграм (LD) для автоматизації роботи підземної автостоянки. Виконайте дії, описані у п.2.2.4.1 теоретичних відомостей.
2. Проведіть емуляцію та налагодження програми у середовищі програмування Zelio Soft. Для цього скористайтесь меню Simulation та натисніть кнопку Run. Після проведення перевірки роботи програми натисніть кнопку Stop.
3. Запишіть створену програму у ПЛК та дослідіть її роботу. Для цього необхідно виконати наступні дії:
 - подати живлення на стенд;
 - перевести контролер у режим STOP;

- увійти у головне меню, натиснувши клавішу Sel./OK;
 - вибрати пункт меню Transfer.\ PC -> Module;
 - у середовищі програмування Zelio Soft вибрати пункт меню Transfer\ Transfer Program\ PC -> Module.
4. Навчитися керувати запуском та зупинкою програми з клавіатури контролера та з комп'ютера. Для керування виконанням програми з клавіатури контролера скористайтеся пунктом меню Run\ Stop, з комп'ютера – пунктами меню середовища Zelio Soft Transfer\ Run Module, Transfer\ Stop Module.
 5. Для відслідковування роботи програми на комп'ютері виберіть пункт меню Mode\ On Line Monitoring Mode.
 6. Складіть пояснюючу таблицю використаних входів і виходів ПЛК, в якій зазначте назву входу-виходу, тип, призначення.

Вимоги до програм:

- Обрахунок кількості машин у гаражі, якщо в'їзд та виїзд з гаража ізольовані. При заповненні автостоянки передбачити світлову сигналізацію. При цьому система повинна закрити в'їзні ворота.
- Передбачити також ручний режим роботи стоянки (обрахунок кількості машин у гаражі, блокування сигналу на закриття входних воріт, коли необхідно пропустити спеціальні служби (пожежних, швидку допомогу тощо)).
- Часова затримки при відкритті та закритті воріт у гаражі, так щоб машина встигла проїхати.
- Заборонити доступ на стоянку у неробочі години і дозволити охороні відключати цю функцію у разі необхідності. Стоянка працює з понеділка по п'ятницю з 8:00 до 19:00, в суботу з 9:00 до 17:00 і повністю зачинена у неділю.
- З метою безпеки необхідно вилучати токсичні викиди, такі як CO_2 , за допомогою вентилятора, коли рівень їх концентрації перевищує допустимий (рівень концентрації викидів визначається за допомогою спеціального датчика, який забезпечує вихідну величину від 0 до 10 В).

- Світло у підземному гаражі повинно автоматично вмикатися по прибуттю автомобіля або за допомогою 2-х кнопочних перемикачів, розташованих біля пішохідного переходу (одна для ескалатора і одна для сходів; через в'їзні ворота пішохідний доступ закритий). З метою економії електроенергії світло повинно автоматично вимикатися через 10 хвилин, за цей час клієнт, як правило, встигає припаркуватися і піднятися ескалатором або взяти машину і залишити автостоянку.

Аналіз задачі:

Таблиця 1.11

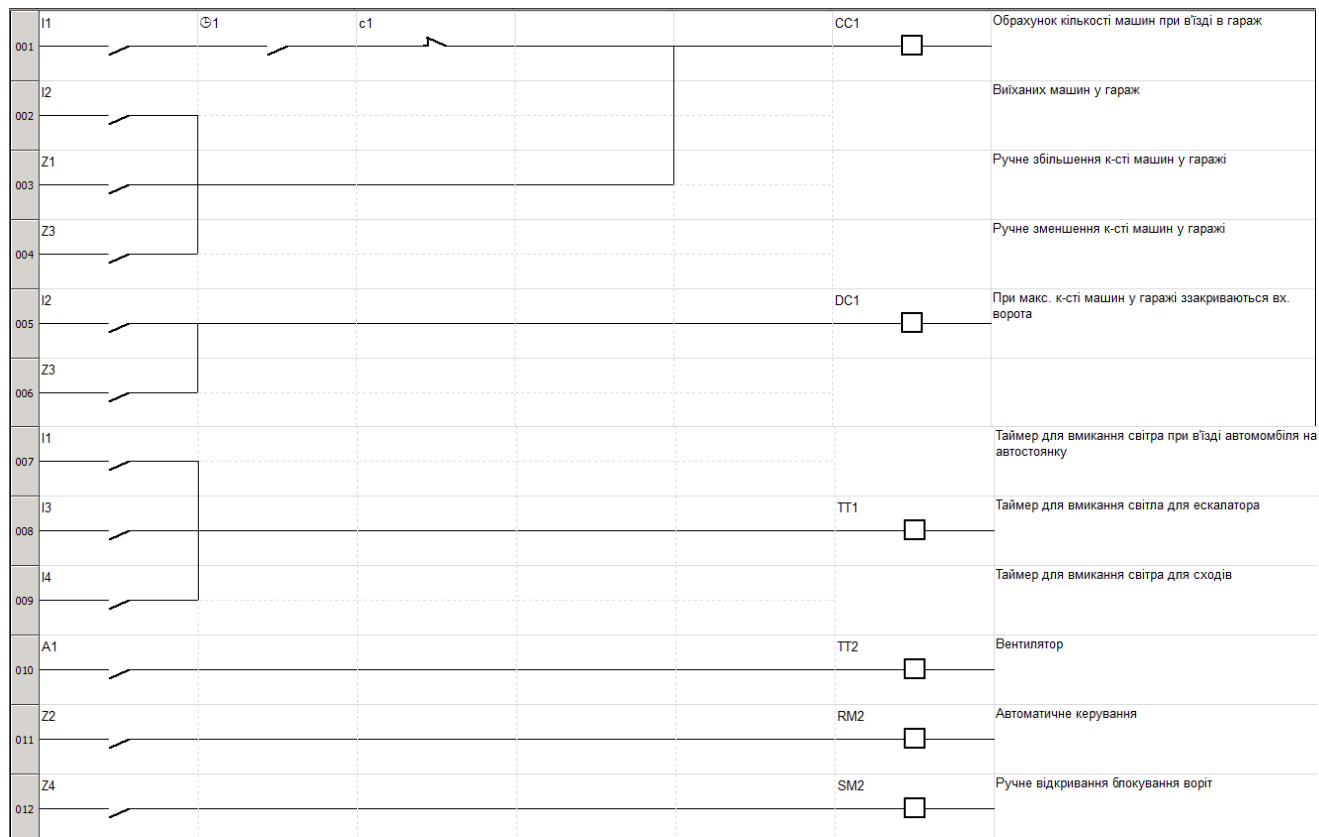
<i>Позначення реле</i>	<i>Опис</i>
Вхід I1	Виявлення в'їзду машини
Вхід I2	Виявлення виїзду машини
Лічильник C1	Підрахунок кількості машин на автостоянці
Вихід Q1	Індикація заповненості стоянки
Вихід Q2	Блокування бар'єру (заборона відкриття бар'єру)
Функціональна клавіша Z4	Ручне блокування бар'єру
Функціональна клавіша Z2	Відновлює автоматичне керування в'їздом
Функціональна клавіша Z1	Ручне збільшення кількості машин на стоянці
Функціональна клавіша Z3	Ручне зменшення кількості машин на стоянці
Функціональний блок Годинник 1	Обробляє години доступу
Входи I3 та I4	Кнопки в точках доступу для пішоходів для включення освітлення автостоянки. Один для підйомника і один для сходів (для пішоходів доступу через в'їзд для машин немає)
Вихід Q3	Керування світлом
Функціональний блок Таймер 1	Таймер для світла (10 хв)

Аналоговий вхід ІВ	Давач рівня CO ₂
Аналоговий функціональний блок А1, порогове значення відповідає 8.5 В	Порівнює значення, видане давачем, зі значенням уставки
Вихід Q4	Керує вентилятором продувки забрудненого повітря
Функціональний блок Таймер 2	Таймер вентилятора (15 хв)

Зверніть увагу, що для реалізації даної задачі необхідний блок Zelio з аналоговими входами, годинником, мінімум 4-ма дискретними входами, мінімум 4-ма дискретними виходами.

Створення програми мовою LD:

Зверніть увагу, що котушки СС і DC повинні з'являтися у програмі лише один раз для одного і того ж лічильника.



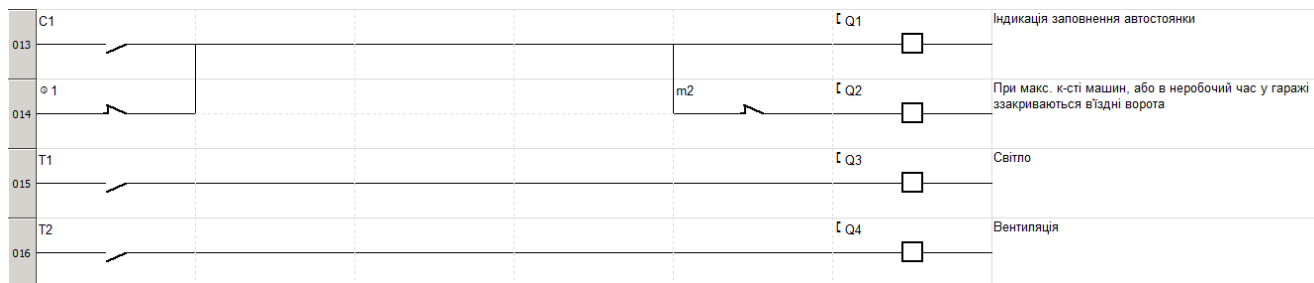
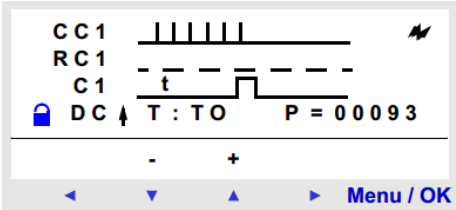
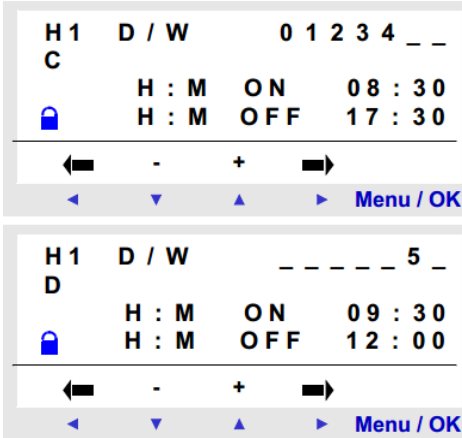
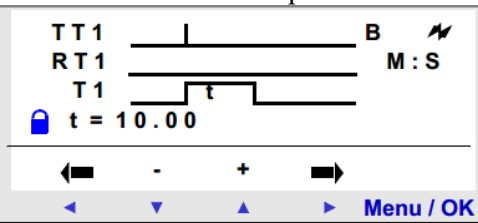
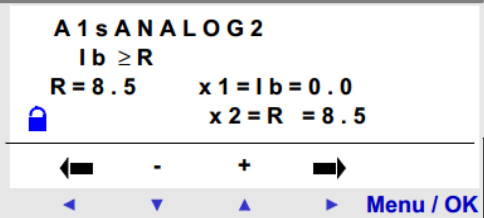
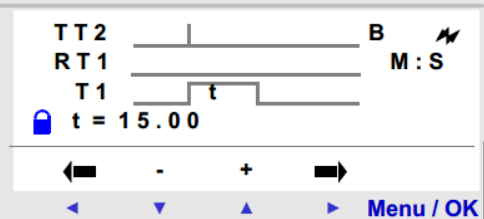


Рис. 1.15 Програма мовою LD

Конфігурування функціональних блоків:

Таблиця 1.12

Функціональний блок	Опис
<p>ФБ Лічильник 1</p> 	<p>Значення уставки = 93 (Максимальне число транспортних засобів на автостоянці. Для проведення емуляції програми це значення доцільно зменшити).</p>
<p>ФБ Годинник 1</p> 	<p>Години роботи: З понеділка по п'ятницю з 8:30 до 17:30. В суботу з 9:30 до 12:00. Неділя зачинено.</p> <p>Використовуються інтервали. 2</p>
<p>ФБ Таймер 1</p> 	<p>Таймер освітлення: 10 хв.</p>

<p style="text-align: center;">Аналоговий ФБ А1</p> 	<p>Порівнює вимірний рівень CO₂ з уставкою: 8.5 В.</p>
<p style="text-align: center;">ФБ Таймер 2</p> 	<p>Тривалість роботи вентилятора при підвищенні порогу по CO₂: 15 хв.</p>

6. Контрольні запитання

1. Яким чином можна охарактеризувати системи контролю та керування типу «Розумний дім»?
2. Яка будова контролера Zelio Logic2?
3. Які основні характеристики контролера Zelio Logic2?
4. Які правила підключення Zelio Logic2?
5. Назвіть пункти та функції головного меню контролера Zelio Logic2.
6. Назвіть пункти та функції меню конфігурації контролера Zelio Logic2.
7. Яким чином запрограмувати контролер мовою драбинкових діаграм (Ladder Diagram Language)?
8. Назвіть основні графічні елементи мови драбинкових діаграм.
9. Яким чином записати програму у ПЛК з комп'ютера? Яким чином зчитати записану програму з ПЛК на комп'ютер?
10. Як можна відслідкувати роботу програми з комп'ютера у реальному часі?

Робота №2. Розробка і випробування системи контролю та керування температурою повітря і вологості ґрунту в теплиці на базі ПЛК Zelio Logic та її випробування мовою LD.

1. Мета роботи

Навчитися розробляти системи контролю та керування температурою повітря і вологості ґрунту в теплиці на базі програмованого логічного контролера. Навчитися програмувати контролер Zelio Logic2 мовою драбинкових діаграм (LD - Ladder Diagram).

2. Теоретичні відомості

2.1 Фактори, що впливають на розвиток та врожайність сільськогосподарських культур

Вода – один з елементів родючості ґрунту. У період росту рослини споживають велику кількість води, з якої 0,01-0,03 % іде на утворення рослинних тканин, решта витрачається на транспірацію (випаровування) листям і стеблами рослин. Сільськогосподарські культури пред'являють певні вимоги до водного режиму ґрунту. Максимальна врожайність досягається лише при оптимальній кількості вологи, живлення, тепла, повітря і світла. При нестачі або надлишку вологи в ґрунті врожайність культур знижується. У першому випадку рослини страждають від нестачі вологи і поживних речовин, у другому – від нестачі повітря у ґрунті. Однією із основних умов, які визначають потребу рослин у воді, являється тривалість критичних фаз розвитку рослин. Розрізняють 4 фази, в яких рослини найбільш чутливі до нестачі вологи в ґрунті: проростання, кущіння, цвітіння і дозрівання. Необхідний для сільськогосподарських культур водний режим ґрунту створюється відповідним режимом зрошення, який встановлює норми, терміни і кількості поливів в залежності від біологічних особливостей культур, природних і господарських умов.

При визначенні витрат води на зрошення сільськогосподарської культури враховують транспірацію води через листя та стебла рослин, випаровування з поверхні ґрунту. Це сумарне випаровування називають **водоспоживанням**, або **евапотранспірацією**. Водоспоживання залежить від кліматичних умов, кількості теплової енергії, яка надходить на поверхню, вологості ґрунту, виду та врожайності культури.

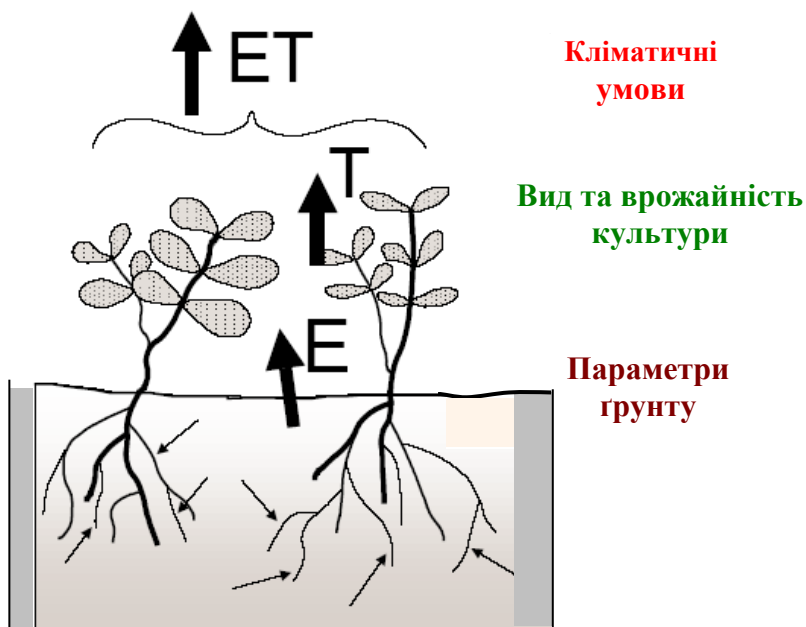


Рис. 2.1. Фактори, що впливають на евапотранспірацію

3. Програма роботи

1. Вивчити будову та характеристики контролера Zelio Logic 2.
2. Дослідити правила підключення контролера Zelio Logic 2.
3. Навчитися користуватися меню контролера Zelio Logic 2
4. Навчитися програмувати контролер Zelio Logic 2 мовою драбинкових діаграм (LD) за допомогою командних клавiш передньої панелі.
5. Навчитися програмувати контролер Zelio Logic 2 мовою драбинкових діаграм (LD) у середовищі Zelio Soft 2.
6. Навчитися записувати програми у контролер.
7. Виконати завдання, поставлені викладачем.

4. Опис лабораторного обладнання

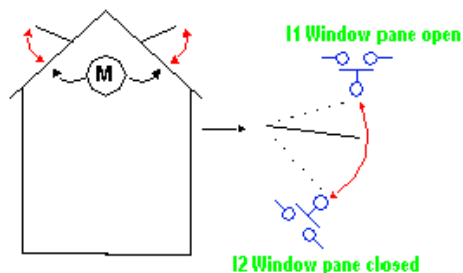
1. Персональний комп'ютер.
2. Операційна система Windows.
3. Лабораторний стенд з логічним контролером Zelio Logic.
4. Середовища програмування Zelio Soft, Zelio Soft 2.

5. Порядок виконання роботи

1. Напишіть програму для програмованого логічного контролера (ПЛК) Zelio Logic мовою драбинкових діаграм (LD) для системи контролю та керування температурою повітря і вологості ґрунту в теплиці. При створенні програми виберіть модель контролера [SR1B122BD Version 1.6](#), середовище для програмування – Zelio Soft.
2. Проведіть емуляцію та налагодження програми у середовищі програмування Zelio Soft. Для цього скористайтеся меню Simulation та натисніть кнопку Run. Після проведення перевірки роботи програми натисніть кнопку Stop.
3. Запишіть створену програму у ПЛК та дослідіть її роботу. Для цього необхідно виконати наступні дії:
 - подати живлення на стенд;
 - перевести контролер у режим STOP;
 - ввійти у головне меню, натиснувши клавішу Sel./OK;
 - вибрати пункт меню Transfer.\ PC -> Module;
 - у середовищі програмування Zelio Soft вибрати пункт меню Transfer\ Transfer Program\ PC -> Module.
4. Навчитися керувати запуском та зупинкою програми з клавіатури контролера та з комп'ютера. Для керування виконанням програми з клавіатури контролера скористайтеся пунктом меню Run\ Stop, з комп'ютера – пунктами меню середовища Zelio Soft Transfer\ Run Module, Transfer\ Stop Module.
5. Для відслідковування роботи програми на комп'ютері виберіть пункт меню Mode\ On Line Monitoring Mode.
6. Складіть пояснюючу таблицю використаних входів і виходів ПЛК, в якій зазначте назву входу-виходу, тип, призначення.

Опис процесу та вимоги до програм:

Власник теплиці замовив установку для автоматичного керування вентиляційними вікнами теплиці, встановленими на даху теплиці, та вологістю ґрунту. Теплиця має 2 вентиляційні вікна. Робота даних вікон керується двигуном і двома давачами положення вікон (відкрито, закрито). На протязі дня вікна відкриваються для вентиляції приміщення з 12:00 до 15:00, коли, як правило, температура найвища. Але, якщо температура є нижчою 10⁰С, вікна не відкриваються, а якщо вони вже відкриті, то закриваються.



Додатково вентиляційні вікна відкриваються на протязі дня, коли температура досягає 25°C . Коли температура падає нижче 25°C , вентиляційні вікна повинні закриватися.

Вночі вентиляційні вікна залишаються закритими незалежно від температури

Отже, у програмі потрібно використати 3 часові діапазони:

- Ніч: з 21:00 до 7:00
- День: з 7:00 до 12:00 та з 15:00 до 21:00
- Полудень: з 12:00 до 15:00

Роботу програми можна представити у вигляді діаграми:

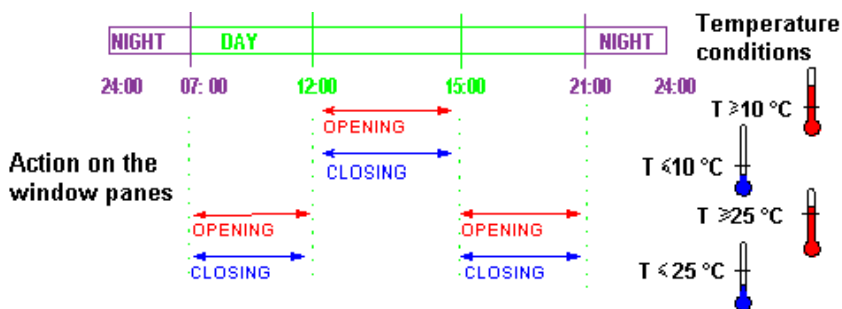


Рис. 2.2 Графік відкриття вентиляційних вікон

Для підтримки заданої вологості ґрунту, яка вимірюється аналоговим давачем вологості, у теплиці передбачений полив дощуванням. Забір води здійснюється насосом із ємності, де вода відстоюється та нагрівається до кімнатної температури. Ємність наповнюється із водопровідної мережі за допомогою електромагнітного клапану і обладнана двома дискретними давачами рівня (максимального та мінімального). Ввімкнення насосу можна проводити, якщо рівень води в ємності не нижчий за мінімальний. Після проведення поливу ємність необхідно заповнити до

максимально допустимого рівня. Для уникнення можливості перезволоження ґрунту тривалість поливу не повинна перевищувати 5 хвилин, а наступний полив можливий тільки через 10 хвилин. Необхідно передбачити також 2 режими поливу: ручний та автоматичний.

No	Contact 1	Contact 2	Contact 3	Contact 4	Contact 5	Coil	Comment
001	☹1	A1	i1			[Q1	відкриті вікна, якщо температура є нижчою 10 град, вікна не відкриваються, а якщо вони вже відкриті, то закриваються
002	☐12-15 ☹2	☐10 град A2					відкриваються вікна, якщо температура вища 25 град.
003	☐7-12 15-21 ☹1	☐25 град A3	i2			[Q2	закриті вікна, якщо температура нижча 10 град.
004	☐12-15 ☹2	☐10 град A4					закриваються вікна, якщо температура нижча 25 град.
005	☐7-12 15-21 ☹3	☐25. град.					вночі закриті вакна, при будь-якій температурі
006	☐21-7 i3					SQ3	постійно під водою
007	i3	i4				RQ3	набір води, коли рівень води в ємності не нижчий за мінімальний i3 та зупинення набору при досяганні i4
008	A5	i3				TT1	
009	☐вимір. вологість ґрунту T1					[Q4	підтримка заданої вологості ґрунту, 3 - імпульс, 5 - пауза

Рис. 2.3 Програма керування теплицею мовою LD

6. Контрольні запитання

1. Які фактори впливають на розвиток та врожайність сільськогосподарських культур
2. Які функціональні блоки були вами використані у програмі для керування температурою повітря та вологістю ґрунту у теплиці? Яке їх призначення?
3. Для чого призначений функціональний блок ГОДИННИК? Яким чином він програмується?
4. Як визначити параметри ПЛК Zelio Logic за його маркуванням?
5. Яким чином запрограмувати логічний контролер Zelio Logic через командні кнопки?
6. Яким чином запрограмувати логічний контролер Zelio Logic через середовище Zelio Soft?
7. Яким чином запрограмувати контролер мовою драбинкових діаграм (Ladder Diagram Language)?
8. Яким чином записати програму у ПЛК з комп'ютера? Яким чином зчитати записану програму з ПЛК на комп'ютер?

Робота №3. Розробка і випробування системи контролю та керування температурою повітря і вологості ґрунту в теплиці на базі ПЛК Zelio Logic та її випробування мовою FBD.

1. Мета роботи

Навчитися розробляти системи контролю та керування температурою повітря і вологості ґрунту в теплиці на базі програмованого логічного контролера. Навчитися програмувати контролер Zelio Logic2 мовою функціональних блоків (FBD – Function Block Diagram), розробляти дворівневі системи контролю та керування на базі Zelio Logic2.

2. Теоретичні відомості

2.1 Програмований логічний контролер Zelio Logic для керування теплицею

Розробимо систему керування температурою та вологістю ґрунту у теплиці на базі програмованого контролера Zelio Logic фірми Schneider Electric. Будову, характеристики, правила підключення, меню програмованого контролера Zelio Logic та правила написання програм мовою драбинкових діаграм розглянуто у роботі 1. Тому розглянемо правила програмування контролера мовою функціональних блоків.

2.1.1 Програмування контролера Zelio Logic2 мовою FBD-блоків

Програми для контролера Zelio Logic2 мовою функціональних блоків (FBD – Function Block Diagram) можна писати, використовуючи спеціалізоване програмне забезпечення Zelio Soft2.

2.1.1.1 Створення нової програми

Для того, щоб створити нову програму необхідно виконати наступні дії.

1. Вибрати меню [File/New](#) або натиснути піктограму [Create a new program](#) під час запуску програми Zelio Soft2. В результаті з'явиться вікно вибору типу модуля.

2. Виберіть потрібний модуль за допомогою ЛК миші. Модулі згруповані за такими ознаками:

- кількість входів-виходів;

- наявність або відсутність дисплею;
- можливість під'єднання шасі розширення.

В результаті здійснення вибору типу модуля з'являється весь список модулів даного типу у вигляді таблиці.

3. Виберіть модуль подвійним натисканням миші або натисніть клавішу **Далее (Next)**. На цьому етапі може виникнути такі ситуації:

- модуль не підтримує можливості розширення та програмується лише мовою LD – перейдіть до кроку 7;
- модуль не підтримує можливості розширення і програмується мовами LD та FBD – перейдіть до кроку 6;
- модуль підтримує розширення, тоді на екрані з'явиться інформація про тип модуля, вибраний у попередньому пункті та типи можливих розширень. Натиснувши кнопку **Add**, можна вибрати необхідне розширення. Для переходу до наступного кроку натисніть кнопку **Далее**. Модуль розширення при необхідності можна видалити кнопкою **Delete**.

4. Виберіть необхідний модуль розширення.

5. Підтвердіть вибір (кнопка **Далее**). В результаті з'явиться вікно вибору мови програмування.

6. Виберіть потрібну мову (LD або FBD). Натисніть кнопку **Далее (Next)**.

7. З'явиться вікно редагування програми.


2.1.3 Мова FBD-блоків (Function Blocks Diagram Language)

При програмуванні мовою FBD-блоків можливе використання 2-х типів вікон:

- вікно редагування;
- вікно супервізора.

Вікно редагування

FBD-програми створюються у вікні редагування. Вікно редагування можна запустити за допомогою меню **Mode/Edit** або за допомогою

піктограми .

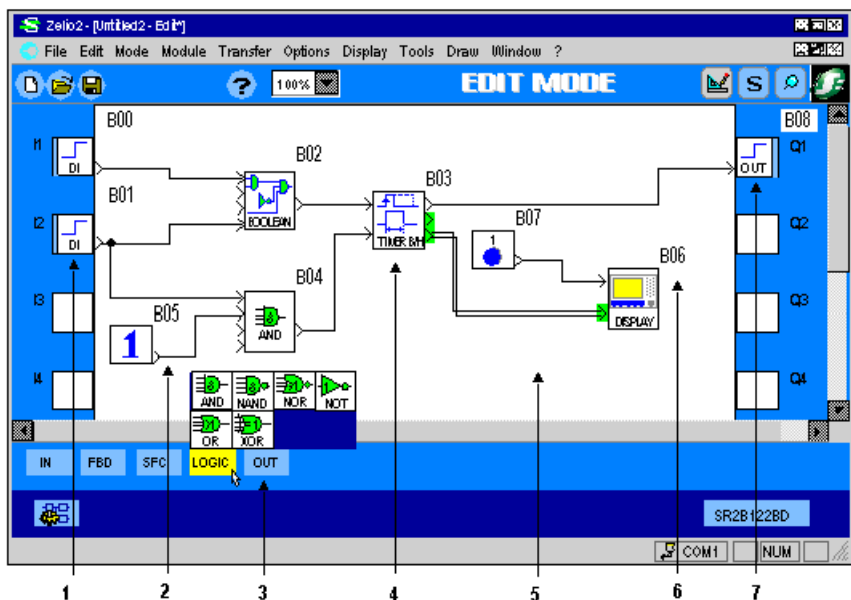




Рис. 3.1 Вікно редагування FBD-програми

Таблиця 3.1

Номер	Опис
1	Входи Zelio
2	З'єднання між 2-ма функціональними блоками
3	Панель функціональних блоків
4	Функціональний блок
5	Вікно програми
6	Номер функціонального блоку
7	Виходи Zelio

Вікно супервізора

Вікно супервізора можна запустити наступним чином:

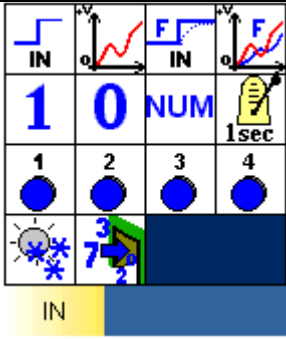
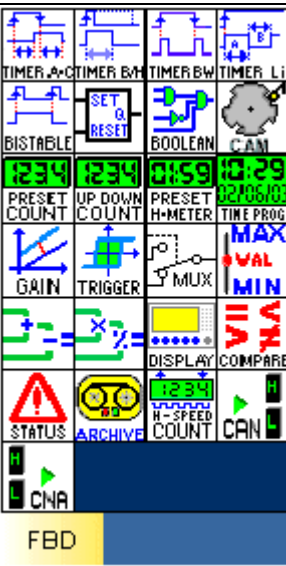
- **симуляція (simulation):** пункт меню [mode/simulation](#) або піктограма ;
- **моніторинг (monitoring):** пункт меню [mode/monitoring](#) або піктограма .

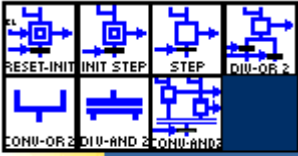



У вікно супервізора можна вставити частини FBD-програми, скопійовані у вікні редагування. У вікні супервізора можна малювати, вставляти текст і малюнки (*.bmp). Доступний також режим емуляції.

Панель елементів

Усі елементи, які необхідні для створення FBD-програми, розміщені на панелі елементів у нижній частині вікна редагування. Елементи розділені на функціональні групи.

Таблиця 3.2

<p>1. Входи</p>	
<p>2. Стандартні функції</p>	

3. SFC-функції	 SFC
4. Логічні функції	 LOGIC
5. Виходи	 OUT
6. Входи-виходи Modbus	

Елементи мови FBD-блоків

1. Входи













Дискретні входи (Discrete-type inputs) –

Тип дискретного входу можна вибрати із вікна [Параметри \(Parameters\)](#).

Таблиця 3.3

Тип	Неактивний стан	Активний стан
Дискретний вхід (discrete input)		
Контакт (contact)		
Вимикач (limit switch)		
Давач наближення (proximity sensor)		

Давач присутності (presence sensor)		
Кнопка з під світкою (illuminated pushbutton)		
Перемикач (selector switch)		
Кнопка (pushbutton)		
Нормально відкрите реле (normally open relay)		



Дискретний вхід з фільтром (Filtered discrete-type input) –

З метою зменшення або навіть ліквідації завад після дискретного входу встановлюють фільтр. Якщо сигнал стабільний на протязі усього визначеного періоду часу, вихід дискретного входу з фільтром приймає виміряне значення, в іншому випадку сигнал лишається незмінним.

Значення параметру (від 1 до 255), введене у вікно [Параметри \(Parameters\)](#), використовується для визначення мінімального часу, на протязі якого сигнал давача повинен бути стабільним. Даний параметр множиться на циклічний час модуля (Zelio).

У режимі симуляції або моніторингу при активації дискретного входу



з фільтром він набуває такого вигляду –







Аналоговий вхід (Analog-type input) –

Аналоговий вхід доступний для всіх модулів постійного струму. Вхідна аналогова напруга конвертується у числове значення від 0 до 255 8-бітним АЦП. Аналоговими входами можуть бути входи від ІВ до ІГ модуля.

За замовчуванням значення аналогового входу лежить в межах 0 – 10 В постійного струму. Тип електричного з'єднання входу можна конфігурувати у вікні [Параметри \(Parameters\)](#). У цьому ж вікні можна вибрати тип аналогового входу.

Таблиця 3.4

Тип	Зображення
Вхід за замовчуванням (Input by default)	
Вхід (Input)	
Температура (Temperature)	
Потенціометр (Potentiometer)	



[Аналоговий вхід з фільтром \(Filtered analog input\)](#) –

Робота такого входу аналогічна аналоговому з тією відмінністю, що після аналогового входу встановлений низькочастотний фільтр.

- Фільтр повністю відновлює вхідний сигнал (частоту, амплітуду та зсув фаз), якщо частота сигналу значно нижча за частоту зрізу фільтра.
- Коли частота вхідного сигналу наближається до частоти зрізу фільтра, вихідний сигнал тієї ж частоти стає меншим за амплітудою і зсунутим за фазою.
- Коли частота вхідного сигналу рівна частоті зрізу фільтра, вихідний сигнал знижується на 30% і зсувається за фазою на 45°.
- Якщо частота вхідного сигналу перевищує частоту зрізу фільтра, вихідний сигнал значно знижується, а зсув за фазою досягає 90°.

У вікні [Параметри \(Parameters\)](#) зазначаються такі параметри аналогового входу:

- вхідна напруга; за замовчуванням – від 0 до 10 В постійного струму;
- частота зрізу фільтра (від 0,06 до 88,25 Гц).



Ціле числове значення (Integer type input) –

Дана функція використовується для введення цілого 16-бітного цілого числа в діапазоні від –32768 до +32767. Цілий числовий вхід може бути присвоєний входам від J9 до JB модулів розширення.

Спеціальні входи у мові FBD (Special inputs in FBD language)

У мові FBD доступні наступні спеціальні входи:

- кнопка;
- дискретні константи;
- числові константи;
- літній час;
- миготіння на протязі 1 секунди.

Дані входи можуть розміщуватися на сторінці FBD-програми.





Кнопки (Button type inputs) –

Входи типу “кнопка” відповідають клавішам на передній панелі Zelio. Дані входи можуть бути вставлені у FBD-діаграму і у режимі симуляції або моніторингу можуть симулювати контакти.

Дискретні константи (Discrete constant-type inputs)

Існує 2 типи дискретних констант:

- 1 - ;
- 0 - .

У режимі емуляції або моніторингу можна використовувати дані входи в інверсному стані. Тоді символи набувають червоного кольору.





Числові константи (Numerical constant-type inputs) –

Числова константа **NUM** представляє собою ціле число у діапазоні від –32768 до +32767. Дана константа може використовуватися для встановлення значення функціональних не з'єднаних входів:



- GAIN;
- COMP IN ZONE;
- TRIGGER.

Значення константи можна встановити у вікні **Параметри (Parameters)**. У режимі емуляції або моніторингу є можливість змінювати константу.

Вхід відображення літнього та зимового часу (Summer Time Input)

Вхід активний на протязі літнього часу –  і неактивний на протязі зимового часу –  .





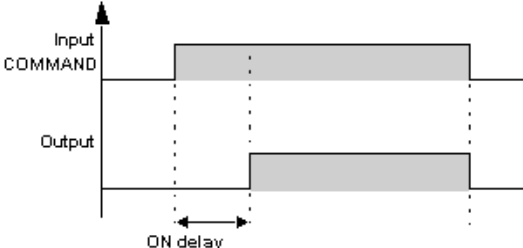
Імпульсний вхід (Blinking input)

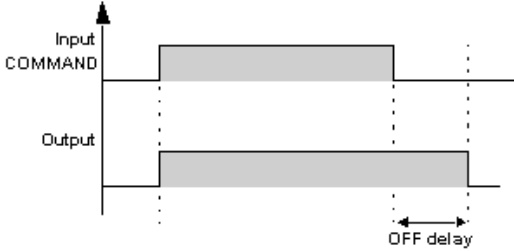
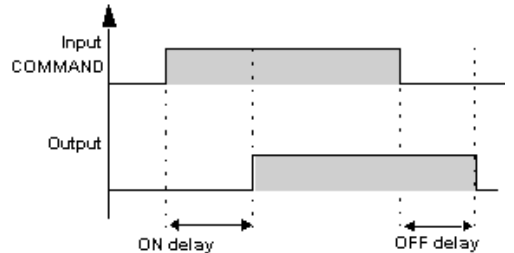


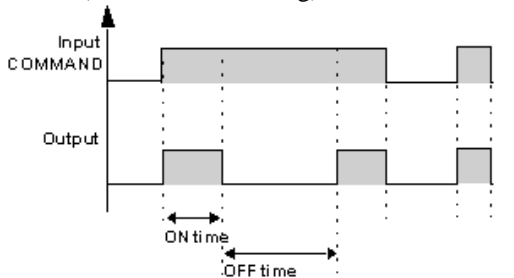
Даний вхід активний кожену секунду. Його активний символ –  , його неактивний символ –  .

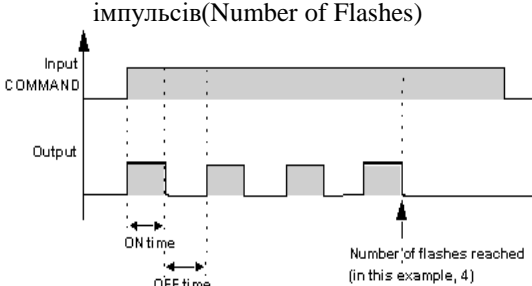
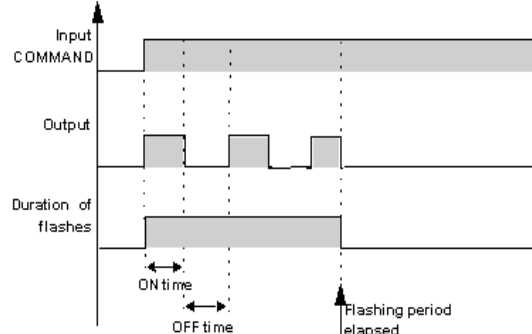

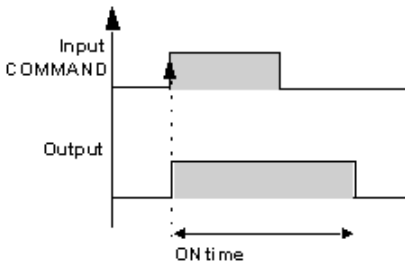
2. Стандартні функції

Таблиця 3.5






Функція	Сим-вол	Пояснення		
BOOLEAN Equation		Вихід приймає значення 1 або 0 в залежності від значень входів згідно таблиці істинності.		
SET-RESET		Вихід Q набуває значення 1 або 0 згідно таблиці істинності:		
		SET	RESET	Q
		0	0	0
		імпульс 0 → 1	0	1
		0	імпульс 0 → 1	0




		1	1	стан виходу залежить від встановленого пріоритету входів
PRESET COUNT Up-Down Counter		Лічильник, який використовується для підрахунку імпульсів від нуля до значення уставки (preset) та від значення уставки до нуля.		
FAST COUNTER		Лічильник, який дозволяє вести підрахунок імпульсів з частотою до 1 КГц. Входи даного лічильника неявно зв'язані з входами I1 (сумуючий вхід) та I2 (декрементуючий вхід) контролера. Тому входи I1 та I2 не рекомендується використовувати для інших цілей при використанні швидкого лічильника. Даний функціональний блок активний лише тоді, коли активний вхід ENABLE. Роботу лічильника не можна симулювати у середовищі програмування.		
UP-DOWN COUNTER		Лічильник, який використовується для підрахунку імпульсів, починаючи від значення уставки, яка вираховується поза межами функціонального блоку. Щоб занести у лічильник значення уставки, необхідно підключити до входу PRESET константу NUM, аналоговий вхід тощо та подати 1 на вхід PRESET FORCING. Вихід лічильника активний, якщо число підрахованих імпульсів дорівнює або більше значення уставки PRESET.		
TIMER A/C		<p>Таймер, який може працювати у трьох режимах:</p> <p>A - затримка включення (on-delay):</p>  <p>C - затримка виключення (off-delay):</p>		



		 <p><i>A/C</i> - комбінований режим:</p> 
TIMER BW		<p>Таймер використовується для генерації імпульсу певної тривалості на виході при подачі імпульсу на вхід таймера. Можливі типи вхідних імпульсів:</p> <ul style="list-style-type: none"> • імпульс по передньому фронту; • імпульс по задньому фронту; • обидва типи імпульсів. <p>При налаштуванні таймера потрібно вибрати один із зазначених типів вхідних імпульсів.</p>
TIMER Li		<p>Таймер працює у режимі Li, який призначений для генерації імпульсів. Є три модифікації даного режиму:</p> <ol style="list-style-type: none"> 1. Тривала генерація імпульсів (Continuous Flashing)  <ol style="list-style-type: none"> 2. Генерація певного числа


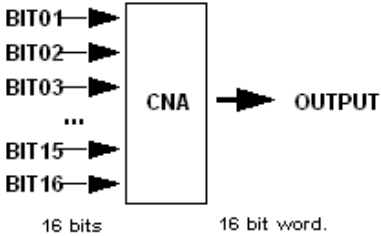

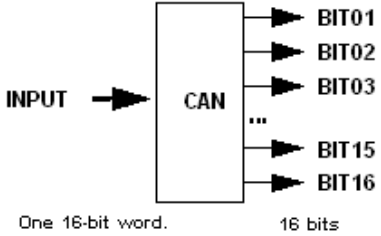
		<p>імпульсів(Number of Flashes)</p>  <p>3. Генерація імпульсів на протязі певного часу (Duration of flashes)</p> 
<p>TIMER B/H</p>		<p>Даний таймер генерує імпульс виході під час приходу переднього фронту імпульсу на вхід. Даний таймер може працювати у двох режимах:</p> <ul style="list-style-type: none"> • <i>B</i>  <ul style="list-style-type: none"> • <i>H</i>

		
COMP IN ZONE Comparison		<p>Функція порівняння використовується для порівняння значення змінної з мінімальним та максимальним значеннями.</p>
PRESET H- METER hour counter		<p>Даний блок призначений для вимірювання проміжку часу, на протязі якого активний командний вхід блоку (Command). Коли період часу активності входу досягає значення уставки, стає активним вихід блоку. Уставка за часом може задаватися в годинах (32757 максимум) та у хвилинах (від 0 до 59). Скидання виходу блоку в нуль відбувається шляхом активації входу Reset.</p>
SCHMITT TRIGGER		<p>Блок тригер Шмітта дозволяє контролювати значення аналогової змінної по відношенню до двох сталих значень – нижньої межі та верхньої межі. Дискретний вихід блоку змінює свій стан на протилежний, якщо:</p> <ul style="list-style-type: none"> • вхідне значення менше значення нижньої межі (min); • вхідне значення більше значення верхньої межі (max). <p>Вихід не змінює свого значення, якщо вхідне значення знаходиться між min і max. Блок працює, тільки якщо активний дискретний вхід <u>Enable</u>.</p>
COMPARE		<p>Блок порівняння призначений для порівняння значень двох аналогових змінних. Вихід блоку активний, якщо результат порівняння змінних <u>Value 1</u> та <u>Value 2</u> позитивний, а дозволяючий вхід <u>Enable</u> активний або ні з чим не з'єднаний. Конкретну операцію порівняння потрібно вибрати у вікні параметрів даного блоку.</p>

GAIN		<p>Функція для масштабування аналогових сигналів за формулою: Calculation Output = A/B * Calculation Input + C</p> <p>Для роботи даного функціонального блоку його вхід <u>Enable Function Input</u> повинен бути активний або ні з чим не з'єднаний.</p>
DISPLAY		<p>Дана функція дозволяє виводити на екран контролера такі дані, як текст (до 72 символів), дату, час, числове значення. У програмі одночасно можуть бути активними до 32 блоків Display. Вхід Enable Function Input має бути активним для запуску блоку. Вхід Value Input визначає дані, які будуть виводитися на екран:</p> <ul style="list-style-type: none"> • вхід ні з чим не з'єднаний – на екран виводитимуться дані, задані користувачем у User options zone при налаштуванні блоку; • вхід з'єднаний з джерелом даних (виходом попереднього функціонального блоку) – на екран виводитимуться дані, які надходять на вхід Value Input.
TIME PROG		<p>Функціональний блок Годинник, який дозволяє задавати інтервали часу у днях тижня, годинах, хвилинах, коли блок активний чи неактивний.</p>
BISTABLE Impulse Relay		<p>Бістабільне імпульсне реле, яке змінює стан свого виходу на протилежний під час приходу запускаючого імпульсу по передньому фронту на його вхід Command. Вхід Reset блоку призначений для скидання його виходу в нуль.</p>
Multiplexing		<p>Функціональний блок мультиплексора, призначений для комутування двох інформаційних входів (Channel A і Channel B) на один вихід. Адресу входу для опитування задає адресний вхід Command. Якщо вхід Command=0 або ні з чим не з'єднаний, то опитується вхід A, якщо Command=1, то опитується вхід B.</p>

ADD-SUB		<p>Функціональний блок для здійснення арифметичних операцій додавання та віднімання трьох цілочисельних даних. Значення виходу блоку розраховується за формулою:</p> <p>Calculation Output = Input 1 + Input 2 – Input 3</p> <p>Якщо вхід блоку Error Propagation=1, розрахунок не проводиться.</p> <p>Вихід Error/Overrun=1, якщо результат операції вийшов за межі відрізка [-32768; +32767] або вхід Error Propagation=1.</p>
MUL-DIV		<p>Функціональний блок для здійснення арифметичних операцій множення та ділення трьох цілочисельних даних. Значення виходу блоку розраховується за формулою:</p> <p>Calculation Output = Input 1 * Input 2 / Input 3</p> <p>Якщо вхід блоку Error Propagation=1, розрахунок не проводиться.</p> <p>Вихід Error/Overrun=1, якщо результат операції вийшов за межі відрізка [-32768; +32767], вхід Error Propagation=1 або вхід Input 3=0.</p>
CAM BLOCK		<p>Функціональний блок кулачкового механізму контролює роботу 8-ми дискретних виходів. На кожному кроці виходи приймають наперед задані значення. Обертання кулачкового механізму контролюють 2 дискретні входи:</p> <ul style="list-style-type: none"> • MOVE FORWARD – перехід на крок вперед (має більший пріоритет); • MOVE BACKWARD – перехід на крок назад. <p>Вхід RESET використовується для ініціалізації функціонального блоку (переведення його на перший крок). Вихід POSITION вказує на номер поточного кроку.</p> <p>При конфігуруванні кулачкового механізму необхідно задати наступні параметри:</p> <ul style="list-style-type: none"> • кількість кроків (number of steps) від 1 до 50; • значення виходів на кожному кроці.


		<div>Number of program steps3</div> <table><tr><th>Step</th><th>W1</th><th>W2</th><th>W3</th><th>W4</th><th>W5</th><th>W6</th><th>W7</th><th>W8</th></tr><tr><td>1</td><td>0</td><td>0</td><td>1</td><td>1</td><td>1</td><td>0</td><td>0</td><td>0</td></tr><tr><td>2</td><td>1</td><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td><td>1</td><td>1</td></tr><tr><td>3</td><td>0</td><td>1</td><td>0</td><td>1</td><td>0</td><td>1</td><td>0</td><td>1</td></tr><tr><td>4</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr></table> <div>Position of the camOutput configuration for each cam positionNumber of selected steps</div>	Step	W1	W2	W3	W4	W5	W6	W7	W8	1	0	0	1	1	1	0	0	0	2	1	1	0	0	0	0	1	1	3	0	1	0	1	0	1	0	1	4	0	0	0	0	0	0	0	0
Step	W1	W2	W3	W4	W5	W6	W7	W8																																							
1	0	0	1	1	1	0	0	0																																							
2	1	1	0	0	0	0	1	1																																							
3	0	1	0	1	0	1	0	1																																							
4	0	0	0	0	0	0	0	0																																							
ARCHIVE		<p>Функціональний блок для запам'ятовування значень двох змінних.</p> <p>Входи блоку ARCHIVE:</p> <ul style="list-style-type: none">• LATCHING – дискретний вхід, при подачі імпульсу на який відбувається запам'ятовування значень змінних;• RESET – дискретний вхід скидання блоку, при цьому запам'ятовані раніше дані втрачаються;• ARCHIVE VALUE 1 – перший вхід для запам'ятовування;• ARCHIVE VALUE 2 – другий вхід для запам'ятовування. <p>Значення змінних запам'ятовуються разом із відомості про час і дату їх надходження.</p> <p>Виходи блоку ARCHIVE:</p> <ul style="list-style-type: none">• VALID ARCHIVE – дискретний вихід, який вказує на наявність запам'ятованих даних;• MINUTE – хвилина (від 0 до 59);• HOUR – година (від 0 до 23);• DAY – день (від 1 до 31);• MONTH – місяць (від 1 до 12);• YEAR – рік (від 0 до 99);• ARCHIVE 1 – значення входу VALUE 1;• ARCHIVE 2 - значення входу VALUE 2.																																													
Module STATUS		<p>Дана функція дозволяє користувачу керувати статусом Zelio 2 і змінювати тим самим поведінку його FBD або SFC програми.</p>																																													

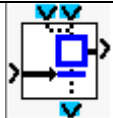

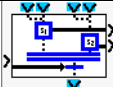
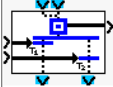

CAN bits to word conversion		<p>Функціональний блок, який перетворює 16-бітний двійковий код у 16-бітне ціле слово.</p> 
CAN word to bits conversion		<p>Функціональний блок, який перетворює 16-бітне ціле слово у 16-бітний двійковий код.</p> 

3. SFC-функції

SFC- функції (Sequential Function Chart) ідентичні функціям мови Grafcet стандарту ІЕС 1131-3. Мова програмування Grafcet призначена для представлення програми у графічній структурованій формі. Діаграма, складена із SFC- функцій, виконується з верху до низу і складається з кроків (Steps) та переходів (Transitions) між ними.

Таблиця 3.6






Функція	Символ	Опис
Initial Step		Початковий крок
Resettable Initial Step		Початковий крок з ініціалізацією за командою.


Step		Крок
AND Divergence		Перехід до двох кроків одночасно
AND Convergence		Перехід від двох паралельно виконуваних кроків до одного
OR Divergence		Перехід від одного кроку до одного або двох кроків
OR Convergence		Перехід від одного або чотирьох кроків до одиничного кроку

4. Логічні функції

У мові програмування FBD є можливість використовувати логічні функції.

Таблиця 3.7

Функція	Символ	Опис	К-сть входів	Тип входів
NO		Вихід активний, якщо вхід неактивний або не підключений і навпаки.	1	Digital
AND		Вихід активний, якщо усі входи активні або не підключені.	4	Digital
OR		Вихід активний, якщо активний хоча б 1 з входів.	4	Digital
NO AND		Вихід активний, якщо хоча б 1 вхід неактивний. Якщо усі входи активні або не підключені, то вихід неактивний.	4	Digital
NO OR		Вихід активний, якщо усі входи неактивні або не підключені. Якщо хоча б 1 вхід активний, то вихід неактивний.	4	Digital

		неактивний.		
EXCLUSIVE OR		Вихід активний, якщо один вхід неактивний, а інший вхід активний або не підключений. Якщо обидва входи активні, неактивні або не підключені, то вихід неактивний.	2	Digital



















5. Виходи















Дискретні виходи (Discrete-type outputs) –

Тип дискретного виходу можна вибрати із вікна [Параметри \(Parameters\)](#).

Таблиця 3.8

Тип	Неактивний стан	Активний стан
Дискретний вихід (descrete output)		
Нормально відкрите реле (normally open relay)		
Лампа (lamp)		
Твердотільне реле (solid state relay)		
Вентиль (valve)		
Поршень (actuator)		
Двигун (motor)		
Резистор (resistance)		
Звуковий сигнал (audible signal)		

Зелена індикаторна лампа (green indicator lamp)		
Червона індикаторна лампа (red indicator lamp)		
Оранжева індикаторна лампа (orange indicator lamp)		
Світловий індикатор (indicator light)		
Нагрівач (heating)		
Вентилятор (fan)		



Цілочисельний вихід (Integer-type output) –

Дана функція використовується для генерування 16-бітного цілого числа у діапазоні від -32768 до +32767. Цілий числовий вихід може бути присвоєний виходам від O9 до JВ модулів розширення.

Вихід підсвітки екрану (LCD Screen Backlighting output)

Дана функція дозволяє програмі контролювати режим підсвічування екрану, вона не може бути присвоєна релейним виходам.



- неактивний стан,



- активний стан.

6. Входи-виходи Modbus

Модуль розширення Modbus **SR3 MBU01BD** може бути підключений до базового модуля Zelio **SR3 BxxxBD**. У FBD режимі 4 (16 бітів) вхідних слова (з J1XT1 по J4XT1) і 4 вихідних слова (з O1XT1 по O4XT1) доступні програмно. Модуль Zelio працює лише у режимі Modbus slave.

Параметри

Параметри встановлюються у робочому середовищі за допомогою пункту меню [Edition\Configuration](#) у режимі [program\Extension MODBUS Tab](#) або при подвійному натисканні на піктограму [Program](#)

[Configuration](#) .

При запуску програми Zelio ініціалізує модуль розширення Modbus.

Модуль має 4 параметри:

- число UART з'єднань і frame формат у мережі Modbus;
- швидкість передачі даних у біт/с;
- UART- парність;
- мережева адреса модуля розширення Modbus.

Входи Modbus

Модуль розширення Modbus SR3 MBU01BD має 4 (16 бітів) входи:

- J1XT1 – шістнадцяткова адреса (0010);
- J2XT1 – 0x0011;
- J3XT1 – %0x0012;
- J4XT1 – %0x0013.

Дані завантажуються з пристрою, що працює у режимі master.

Виходи Modbus

Модуль розширення Modbus SR3 MBU01BD має 4 (16 бітів) виходи:

- O1XT1 – шістнадцяткова адреса (0000);
- O2XT1 – 0x0001;
- O3XT1 – 0x0002;
- O4XT1 – %0x0003.

Дані завантажуються у пристрій, що працює у режимі master.

3. Програма роботи

1. Вивчити будову та характеристики контролера Zelio Logic 2.
2. Дослідити правила підключення контролера Zelio Logic 2.
3. Навчитися користуватися меню контролера Zelio Logic 2
4. Навчитися програмувати контролер Zelio Logic 2 мовою FBD-блоків у середовищі Zelio Soft 2.
5. Навчитися записувати програми у контролер.
6. Навчитися створювати вікно супервізора у середовищі Zelio Soft2.
7. Виконати завдання, поставлені викладачем.

4. Опис лабораторного обладнання

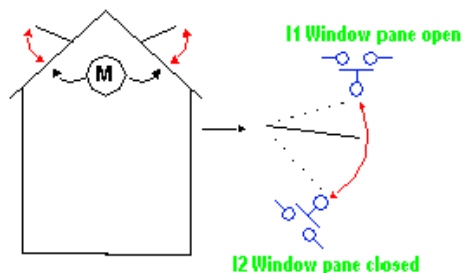
1. Персональний комп'ютер.
2. Операційна система Windows.
3. Лабораторний стенд з логічним контролером Zelio Logic.
4. Середовища програмування Zelio Soft, Zelio Soft 2.

5. Порядок виконання роботи

1. Напишіть програму для програмованого логічного контролера (ПЛК) Zelio Logic2 мовою FBD-блоків для системи контролю та керування температурою повітря і вологості ґрунту в теплиці. При цьому використовуйте середовище програмування Zelio Soft2.
2. Створіть вікно супервізора, в якому намалюйте ФСА технологічного процесу. Для того, щоб переключитися у вікно супервізора скористайтеся меню Window\Supervision.
3. Проведіть емуляцію та налагодження програми у середовищі програмування Zelio Soft2. Дослідіть роботу програми у середовищі програмування Zelio Soft2. При цьому скористайтеся керуючими кнопками, наведеними у таблиці 9.
4. Складіть пояснюючу таблицю використаних входів і виходів ПЛК, в якій зазначте назву входу-виходу, тип, призначення.
5. Зробіть висновки щодо можливостей двох мов програмування – LD та FBD – для розробки програмного забезпечення для керування технологічним процесом.

Опис процесу та вимоги до програм:

Власник теплиці замовив установку для автоматичного керування вентиляційними вікнами теплиці, встановленими на даху теплиці, та вологістю ґрунту. Теплиця має 2 вентиляційні вікна. Робота даних вікон керується двигуном і двома давачами положення вікон (відкрито, закрито). На протязі дня вікна відкриваються для вентиляції приміщення з 12:00 до 15:00, коли, як правило, температура найвища. Але, якщо температура є нижчою 10⁰С, вікна не відкриваються, а якщо вони вже відкриті, то закриваються.



Додатково вентиляційні вікна відкриваються на протязі дня, коли температура досягає 25°C . Коли температура падає нижче 25°C , вентиляційні вікна повинні закриватися.

Вночі вентиляційні вікна залишаються закритими незалежно від температури

Отже, у програмі потрібно використати 3 часові діапазони:

- Ніч: з 21:00 до 7:00
- День: з 7:00 до 12:00 та з 15:00 до 21:00
- Полудень: з 12:00 до 15:00

Роботу програми можна представити у вигляді діаграми:

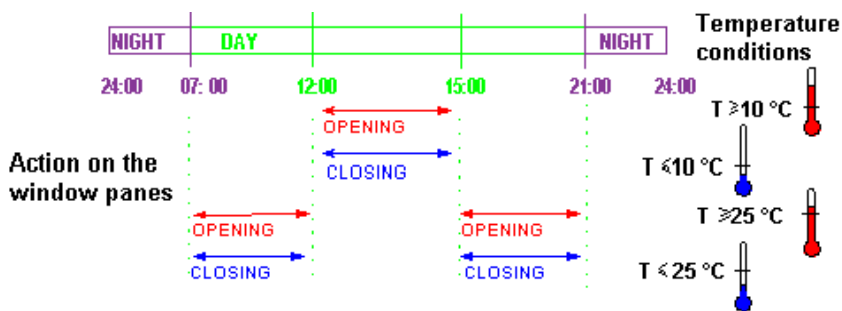
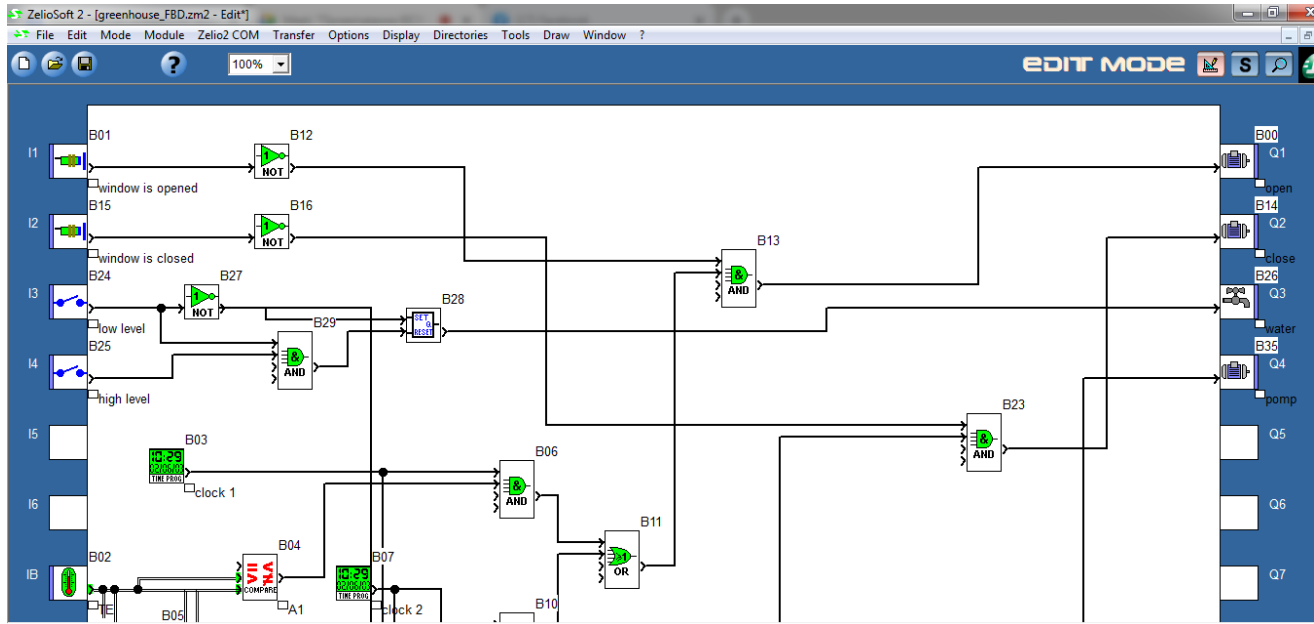
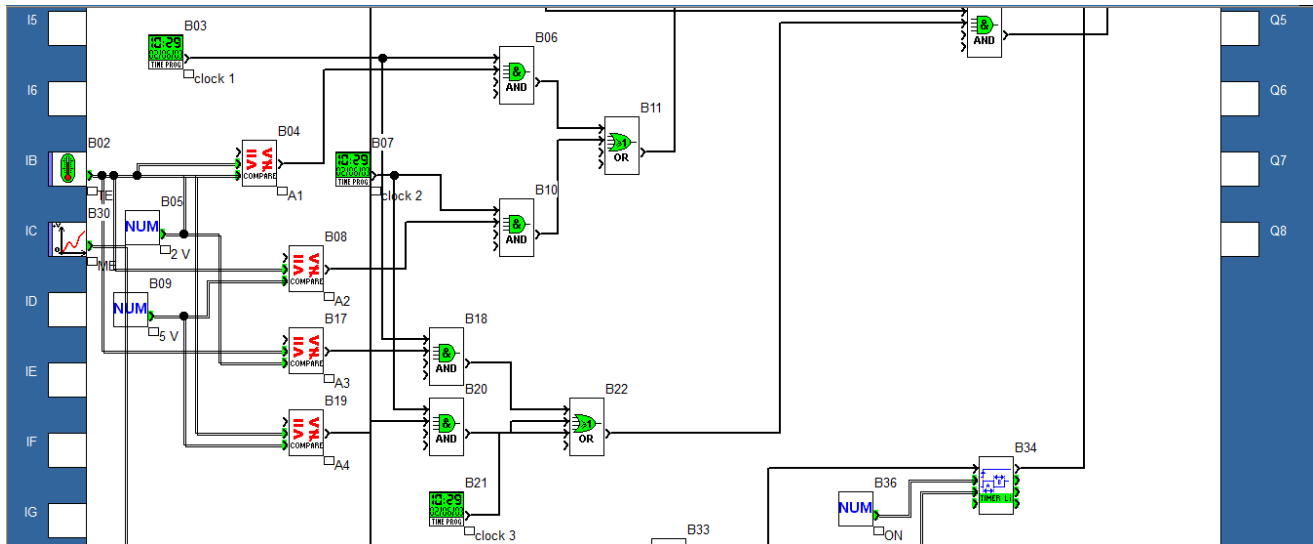


Рис. 3.2 Графік відкриття вентиляційних вікон

Для підтримки заданої вологості ґрунту, яка вимірюється аналоговим давачем вологості, у теплиці передбачений полив дощуванням. Забір води здійснюється насосом із ємності, де вода відстоюється та нагрівається до кімнатної температури. Ємність наповнюється із водопровідної мережі за допомогою електромагнітного клапану і обладнана двома дискретними давачами рівня (максимального та мінімального). Ввімкнення насосу можна проводити, якщо рівень води в ємності не нижчий за мінімальний. Після проведення поливу ємність необхідно заповнити до

максимально допустимого рівня. Для уникнення можливості перезволоження ґрунту тривалість поливу не повинна перевищувати 5 хвилин, а наступний полив можливий тільки через 10 хвилин. Необхідно передбачити також 2 режими поливу: ручний та автоматичний.





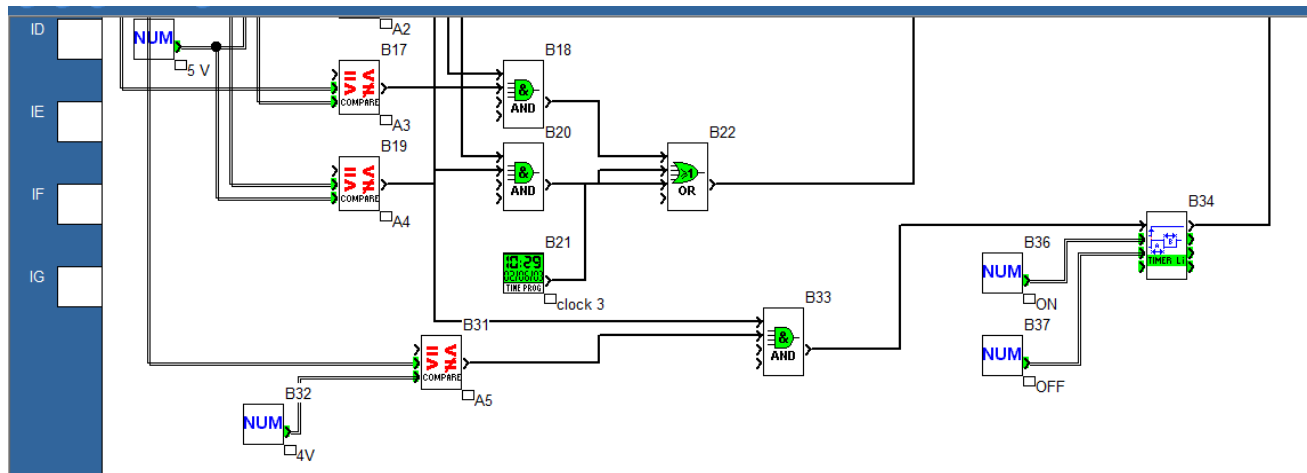


Рис. 3.3 Програма керування теплицею мовою FBD

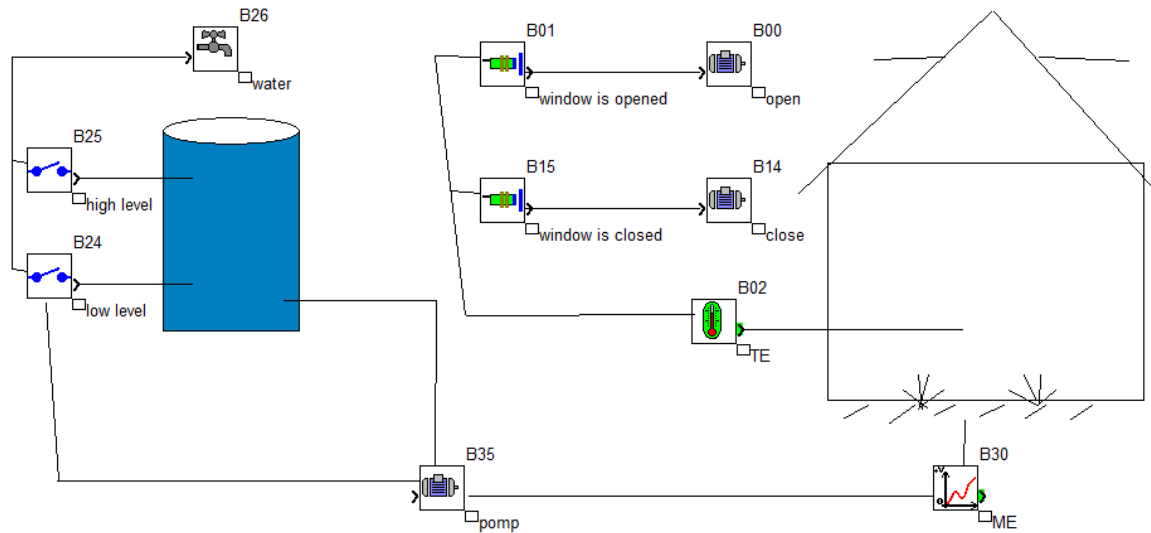


Рис. 3.4 Экран супервізора для моніторингу процесу у режимі реального часу

6. Контрольні запитання

1. Які фактори впливають на розвиток та врожайність сільськогосподарських культур?
2. Які функціональні блоки були вами використані у програмі для керування температурою повітря та вологістю ґрунту у теплиці? Яке їх призначення?
3. Для чого призначений функціональний блок ГОДИННИК? Яким чином він програмується?
4. Як визначити параметри ПЛК Zelio Logic за його маркуванням?
5. Яким чином запрограмувати логічний контролер Zelio Logic через середовище Zelio Soft?
6. Яким чином запрограмувати контролер мовою FBD-блоків (Function Blocks Diagram Language)?
7. Яким чином записати програму у ПЛК з комп'ютера? Яким чином зчитати записану програму з ПЛК на комп'ютер?
8. Як можна відслідкувати роботу програми з комп'ютера у реальному часі?
9. Для автоматизації яких процесів слід використовувати ПЛК Zelio Logic? Які можливості має ПЛК Zelio Logic для роботи у дворівневих системах контролю та керування?
10. Порівняйте можливості двох мов програмування – LD та FBD – при написанні програм для керування температурою повітря та вологістю ґрунту у теплиці.

Робота №4. Розробка і випробування системи контролю та керування на базі ПЛК Siemens Logo.

1. Мета роботи

Навчитися розробляти системи контролю та керування процесами на базі програмованого логічного контролера. Вивчити будову, правила підключення, характеристики, меню контролера Siemens Logo. Навчитися програмувати контролер Siemens Logo мовою функціональних блоків (FBD – Function Block Diagram).

2. Теоретичні відомості

Logo – це інтелектуальне реле фірми Siemens (рис.4.1).



Рис. 4.1 Зовнішній вигляд Siemens Logo

Logo пропонує рішення різноманітних технологічних задач, у тому числі в електрообладнання житлових приміщень (освітлення сходових клітин, зовнішнє освітлення, жалюзі і ролети, освітлення вітрин магазинів тощо), у комутаційних шафах, в управлінні машинами і апаратами (системи керування воротами, вентиляційні системи, насоси для господарчої води). Logo також можна використовувати для систем керування в оранжереях і теплицях, для попередньої обробки сигналів керування та, при підключенні комунікаційного модуля ASi, для децентралізованого керування машинами і процесами. Також є спеціальні варіанти для

без дисплея для серійних проектів у мікромашинобудуванні, апаратобудуванні, шафах керування.

Маркування моделей Logo:

- 12: варіант на 12В постійного струму,
- 24: варіант на 24В постійного струму,
- 230: варіант на 115/240 В змінного струму,
- R: релейні виходи (без R – транзисторні виходи),
- C: вбудований годинниковий вимикач на 7 днів,
- o: варіант без дисплея,
- DM: цифровий модуль,
- AM: аналоговий модуль,
- FM: функціональний модуль (наприклад, ASi).

3. Програма роботи

1. Вивчити будову та характеристики контролера Siemens Logo.
2. Дослідити правила підключення контролера Siemens Logo.
3. Навчитися користуватися меню контролера Siemens Logo
4. Навчитися програмувати контролер Siemens Logo мовою функціональних блоків (FBD) за допомогою командних клавіш передньої панелі.
5. Навчитися програмувати контролер Siemens Logo мовою функціональних блоків (FBD) у середовищі Logo Soft.
6. Навчитися записувати програми у контролер.
7. Виконати завдання, поставлені викладачем.

4. Опис лабораторного обладнання

1. Персональний комп'ютер.
2. Операційна система Windows.
3. Середовище програмування LOGO Soft.

5. Порядок виконання роботи

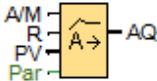
Завдання 1

Написати програму для ПІ регулювання для одноконтурної САР.
При цьому скористатися програмою Logo Soft.

1.1 Написати програму таким чином, щоб завдання для ПІ регулятора було сталою величиною.

Властивості ПІ регулятора у програмному забезпеченні Logo Soft:

PI controller



Таблиця 4.1

Вхід-Вихід	Опис
A/M	Режим роботи регулятора (дискретний сигнал): 1 – автоматичний режим, 0 – ручний режим
R	Режим Reset – скидає вихід регулятора до нуля (дискретний сигнал)
PV	Вхідне виміряне значення параметру (аналоговий сигнал)
Parameter	Налаштування ПІ регулятора: Sensor: тип датчика; Min: мінімальне значення для PV (-10000...+20000); Max: максимальне значення для PV (-10000...+20000); Gain=-10.00...+10.00; Offset=-10.000...+10.000; SP: значення уставки ((-10000...+20000); Mq: значення виходу AQ регулятора в ручному режимі (0...1000); Parameter Sets: уставки ПІ регулятора: KC: коефіцієнт пропорційності (00.00...99.99); Ti: час інтегрування (00.01 хв ... 99.59 хв); Dir: напрям дії регулятора (+ або -); p: кількість знаків після коми (0, 1, 2, 3).
AQ	Вихід ПІ регулятора (аналоговий сигнал)

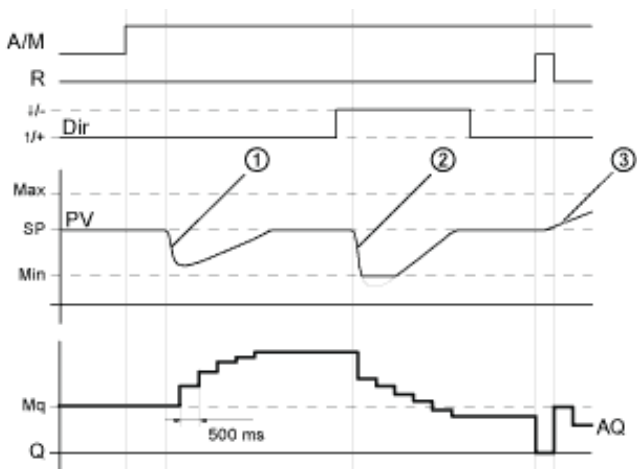


Рис. 4.2 Приклад часової діаграми роботи ПІ регулятора

Для полегшення налаштування ПІ регулятора користувачу одразу пропонується набір деяких типових налаштувань. У той же час лишається можливість ввести усі налаштування вручну. Серед типових налаштувань є такі: швидке керування температурою, повільне керування температурою, керування тиском (1 і 2), керування рівнем (1 і 2). Для даних типових рішень користувачу пропонуються певні значення КС та Ті.

Отже, збираємо наступну програму:

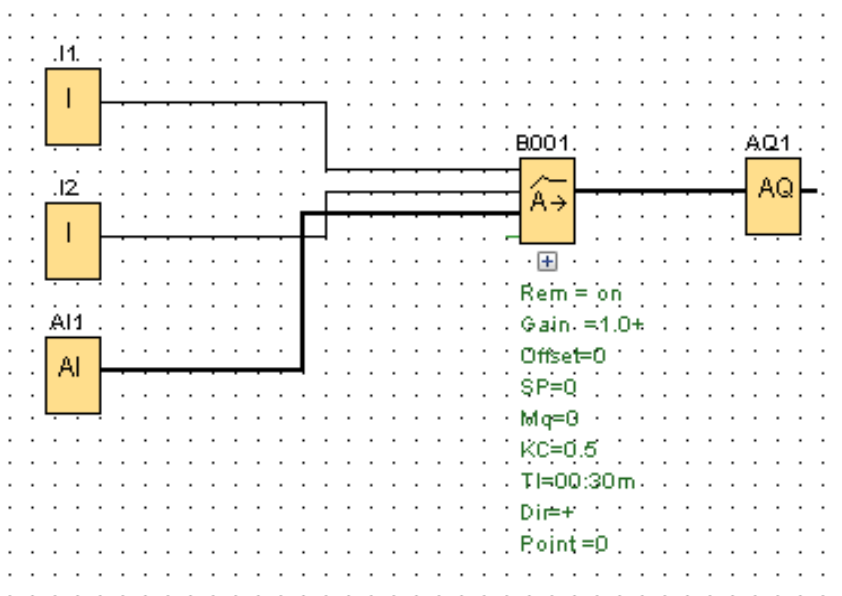


Рис. 4.3 Програма до завдання 1.1

Налаштовуємо ПІ регулятор у вікні налаштувань. **В кінці обов'язково натискаємо кнопку ОК.**

Моделюємо роботу програми. Під час моделювання дослідіть роботу регулятора у різних режимах (автоматичний, ручний) та при різних значеннях вхідного сигналу PV.

B001 [PI controller]

Parameter | Comment

Parameter

Block name:

Sensor

Sensor:

Analog settings

Measurement Range	Parameter
Minimum: <input type="text" value="0"/>	Gain: <input type="text" value="1.00"/>
Maximum: <input type="text" value="1000"/>	Offset: <input type="text" value="0"/>

Output

Set value (SP)

Manual output (Mq)

Parameter:

Parameter set:

Controller amplification (KC):

☒ Integration time (TI): : Minutes (m:s)

Direction (Dir): ☒ Upwards (+)
☐ Downwards (-)

Decimal places

Decimal places in the message text: +12345

Others

☒ Retentivity

☐ Protection Active

OK Cancel Help

Рис. 4.4 Приклад налаштування ПІ регулятора

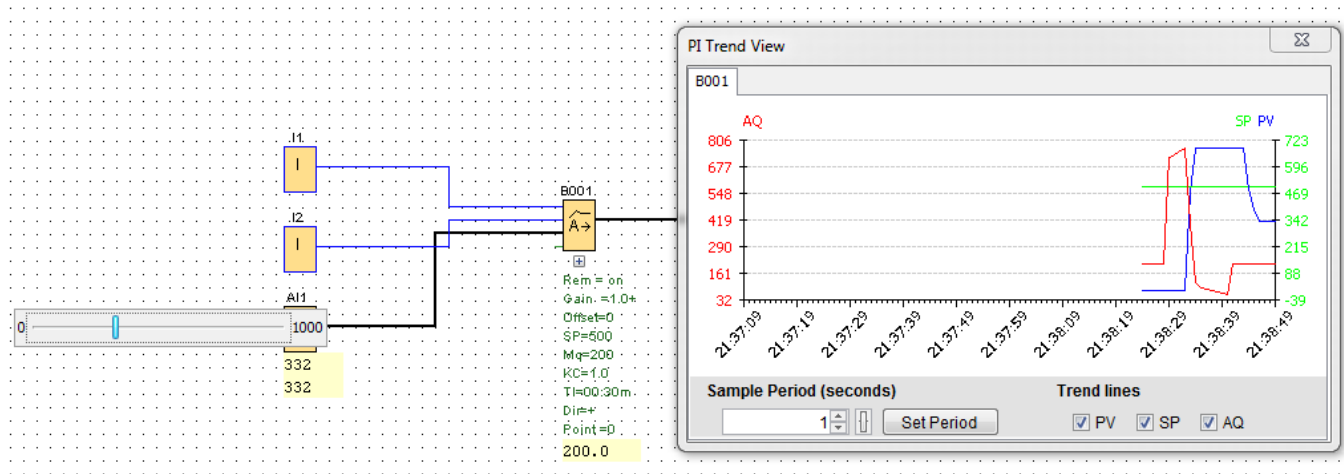


Рис. 4.5 Робота програми у режимі моделювання

1.2 Модернізувати програму таким чином, щоб завдання для ПП регулятора можна було задавати у режимі реального часу.
Для цього використаємо блок Analog Amplifire:

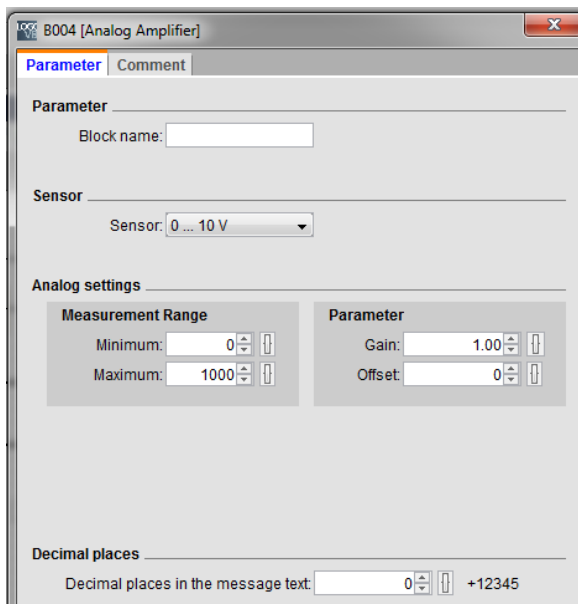


Рис. 4.6 Налаштування блоку Analog Amplifier

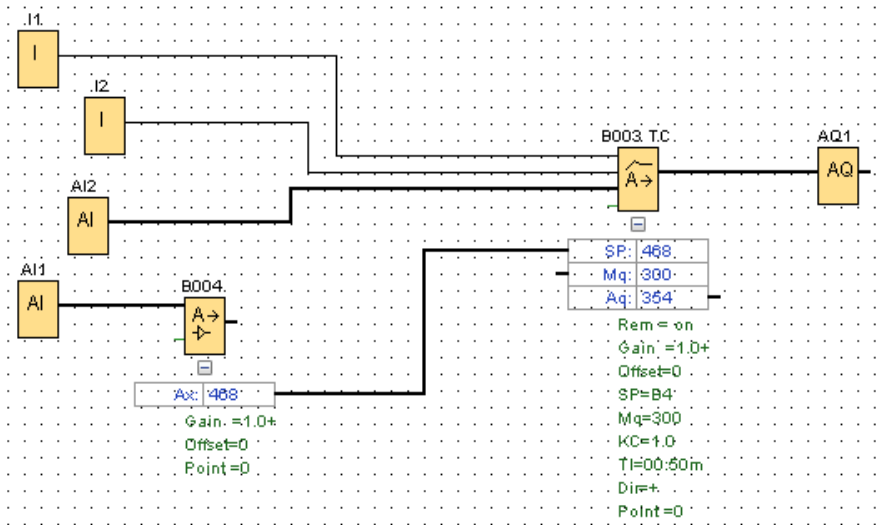


Рис. 4.7 Програма до завдання 1.2

Завдання 2

Написати програму для керування роботою автоматичних дверей у супермаркет згідно нижче наведених вимог.

Вимоги до програми:

Коли хтось наближається, двері повинні автоматично відчинитися. Двері мають залишатися відчиненими, поки хто-небудь ще знаходиться в дверях. Якщо у дверному проході більше нікого немає, двері мають автоматично зачинитися через короткий проміжок часу.

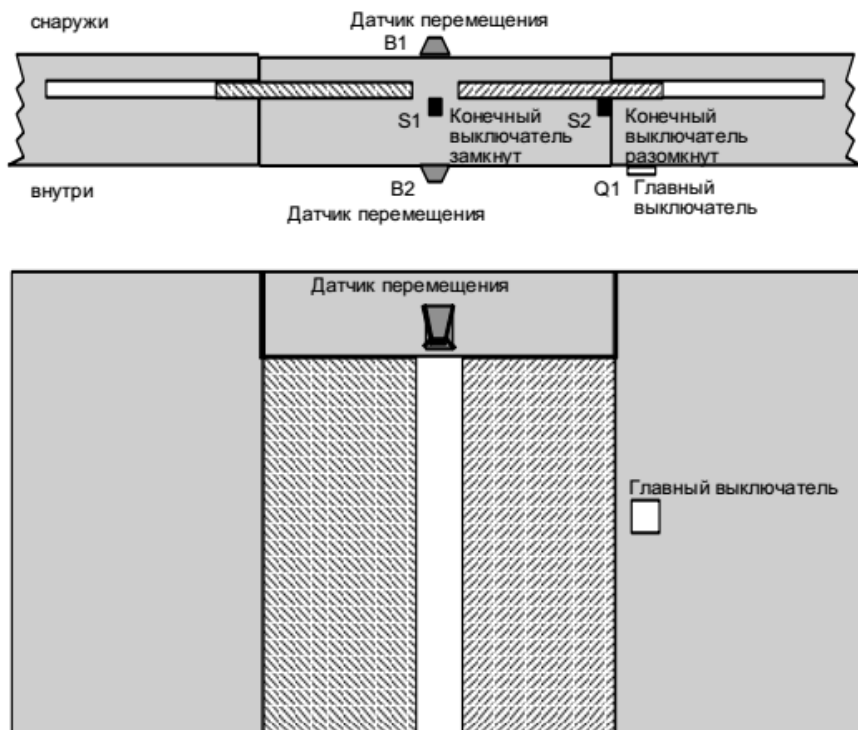


Рис. 4.8 Схема входних дверей у супермаркет

Двері зазвичай приводяться в рух двигуном із запобіжною фрикційною муфтою. Це запобігає тому, щоб людина могла бути затисненою або пораненою дверима. Система керування підключена до мережі через головний вимикач.

Функціональні можливості і зручності для користувача можуть бути покращені наступними способами:

Можна підключити додатковий керуючий перемикач S3 з позиціями: Відчинено – Автоматика – Зачинено (В-А-З).

Можна підключити зумер до одного із виходів ПЛК для попередження про зачинення дверей, що залежить від часу і напрямку (наприклад, відкриття тільки у робочі години; по завершенні роботи відкриття тільки із середини).

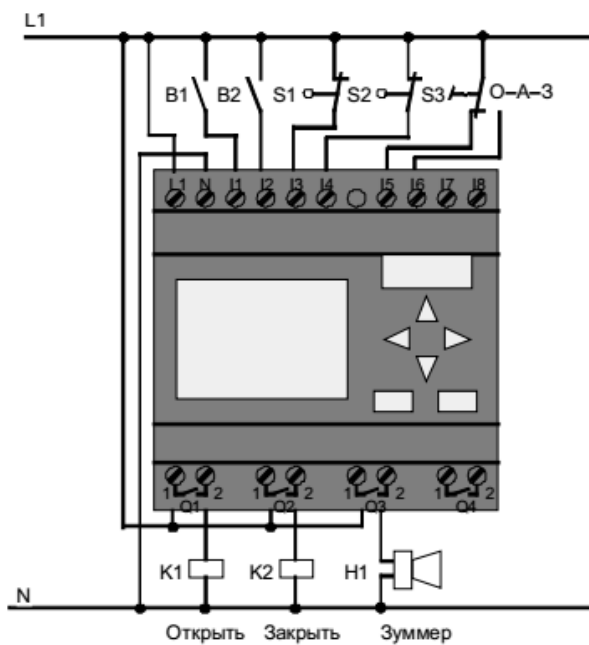


Рис. 4.9 Підключення розширеного рішення з ПЛК Logo

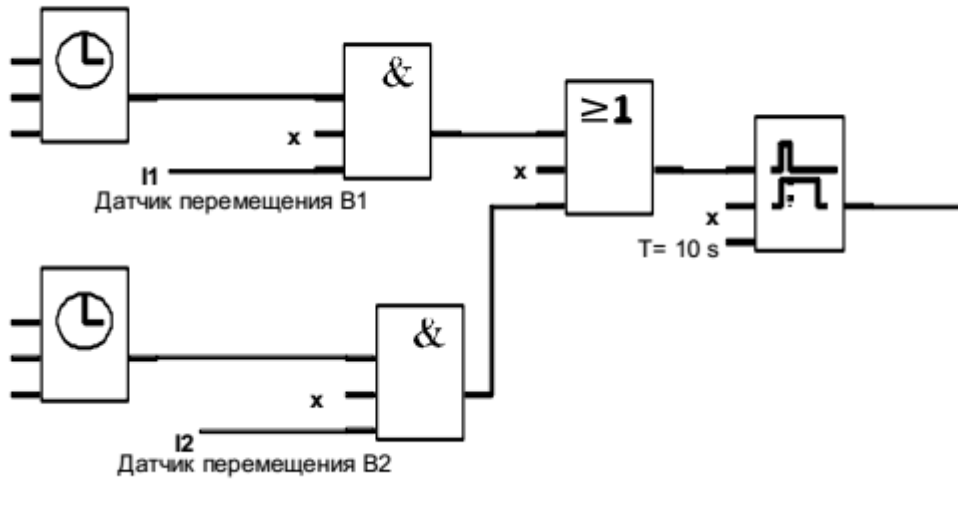
No1:
 День = Пн..Пт
 Вкл = 09:00
 Выкл = 18:00

No2:
 День = Сб
 Вкл = 08:00
 Выкл = 13:00

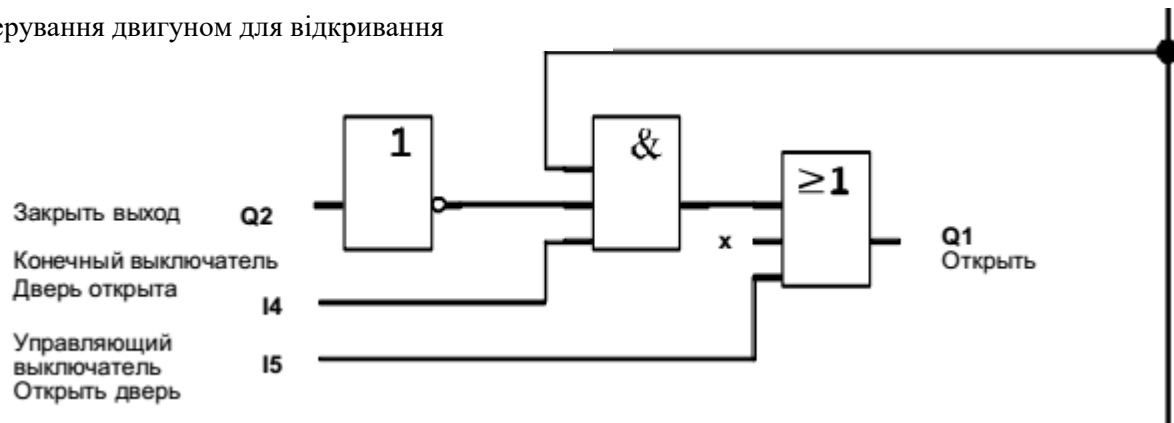
No1:
 День = Пн..Пт
 Вкл = 09:00
 Выкл = 19:00

No2:
 День = Сб
 Вкл = 08:00
 Выкл = 14:00

Виявлення переміщення



Керування двигуном для відкривання



Керування двигуном для закривання

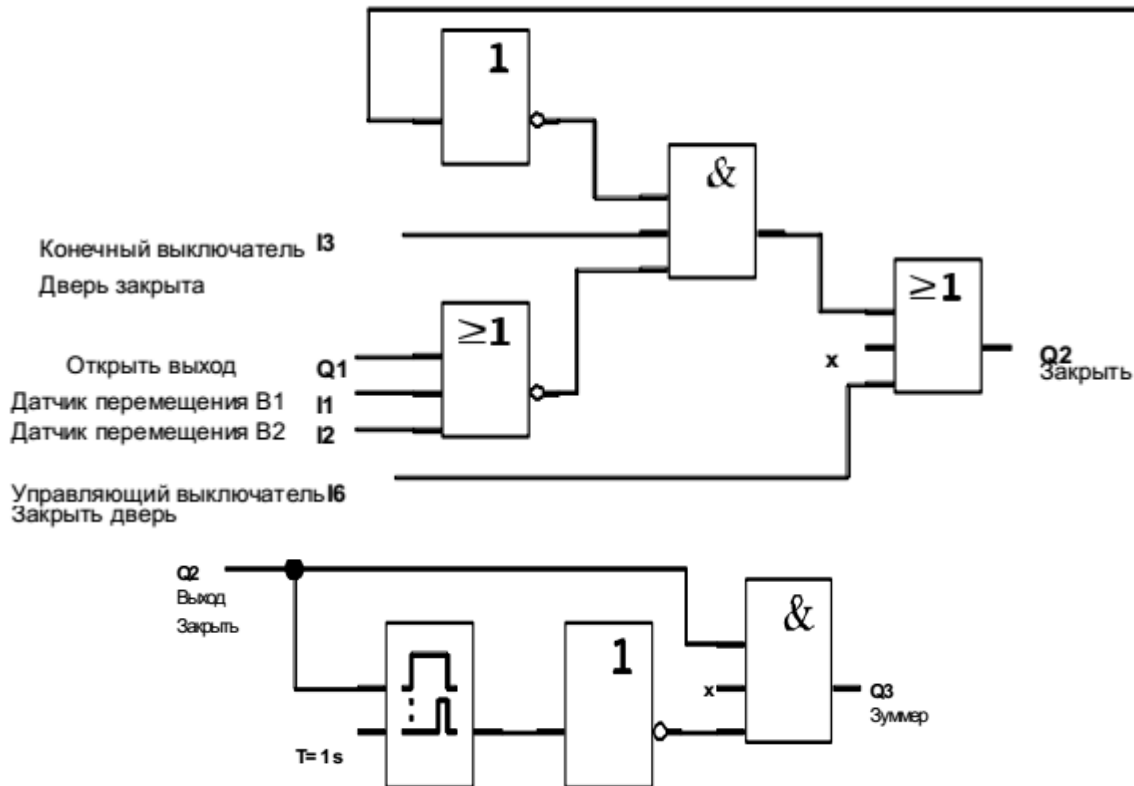
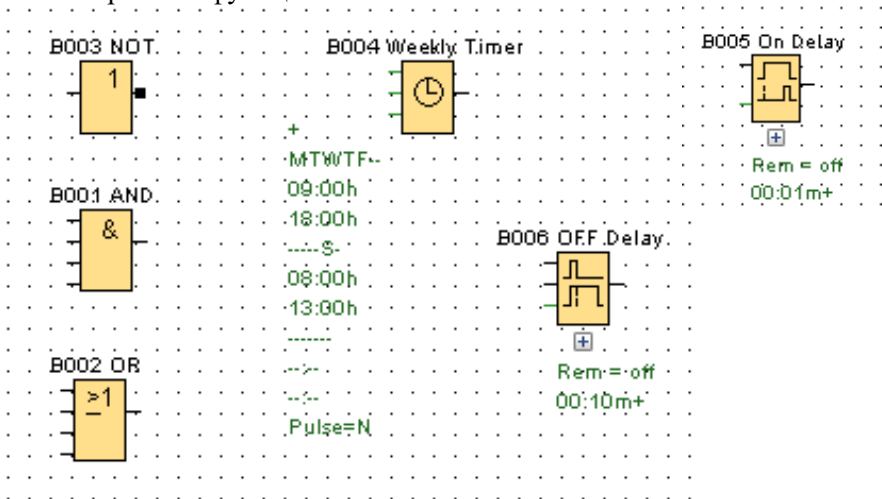


Рис. 4.10 Програма до завдання 2

Використані функціональні блоки:



Виявлення переміщення

У робочий час давач переміщення В1 ініціює відчинення дверей, як тільки хто-небудь хоче увійти у магазин ззовні. Давач переміщення В2 ініціює відчинення дверей, якщо хтось хоче вийти з магазину. По завершенні робочого часу давач переміщення В2 продовжує відчиняти двері протягом 1 години, щоб покупці могли покинути магазин.

Керування двигуном для відчинення дверей

Вихід Q1 включається і відчиняє двері, коли відбувається наступне:

- Працює керуючий вимикач на I5=1 (двері мають бути постійно відчинені) АБО
- Давачі переміщення виявляють, що хтось наближається до дверей (I1=1 АБО I2=1) І
- Двері ще не повністю відчинені (кінцевий вимикач на I4=1).

Керування двигуном для зачинення

Вихід Q2 включається і зачиняє двері, коли відбувається наступне:

- Працює керуючий вимикач на I6=1 (двері повинні бути постійно зачинені) АБО
- Давачі переміщення виявляють, що біля дверей нікого немає (I1=0 І I2=0) І

- Двері ще не повністю зачинені (кінцевий вимикач I3=1).

Зумер

Зумер підключений до виходу Q3. Зумер звучить короткий час (у даному випадку 1 с), коли двері зачиняються.

6. Контрольні запитання

1. Що таке ПІ регулятор, які його переваги у порівнянні із п регулятором?
2. Які налаштування має ПІ регулятор, як вони впливають на його роботу?
3. Яка будова ПЛК Siemens Logo? Які його можливості, області використання?
4. Які функціональні блоки були використані у програмі до завдання 1?
5. Які функціональні блоки були використані у програмі до завдання 2?
6. Яким чином запрограмувати контролер мовою FBD-блоків (Function Blocks Diagram Language)?
7. Яким чином записати програму у ПЛК з комп'ютера? Яким чином зчитати записану програму з ПЛК на комп'ютер?
8. Як можна відслідкувати роботу програми з комп'ютера у реальному часі?
9. Для автоматизації яких процесів слід використовувати ПЛК Siemens Logo?
10. Порівняйте можливості ПЛК Siemens Logo із іншими, відомими вам ПЛК.